

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

L'utilisation des interfaces tangibles pour l'apprentissage des concepts de programmation chez les jeunes

Bodart, Antoine

Award date:
2018

Awarding institution:
Université de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

UNIVERSITÉ DE NAMUR
Faculté d'informatique
Année académique 2017–2018

**L'utilisation des interfaces tangibles pour
l'apprentissage des concepts de
programmation chez les jeunes**

Bodart Antoine



Promoteur : _____ (Signature pour approbation du dépôt - REE art. 40)
Dumas Bruno

Co-promoteur : Julie Henry

Mémoire présenté en vue de l'obtention du grade de
Master en Sciences Informatiques.

Table des matières

1	Introduction	4
2	Contexte	6
3	Problématique et méthodologie de recherche	8
4	Etat de l'art	11
4.1	Interface tangible utilisateur	11
4.1.1	Définition, utilisation et illustration	11
4.1.2	Taxonomies	16
4.1.3	Les TUI et l'apprentissage chez les jeunes	17
4.2	Les concepts de base de la programmation	20
5	Contributions personnelles	22
5.1	Classifications	45
5.1.1	Classification selon Yu [40]	45
5.1.2	Classification selon Fishkin	46
5.1.3	Classification selon un tableau de critères originaux	47
6	Analyse et discussion	52
7	Conclusion	58
8	Travaux futurs	60

Remerciements

Je tiens à remercier toutes les personnes qui m'ont épaulé à la réalisation de ce mémoire. Avant tout, je souhaite remercier Monsieur Bruno Dumas, professeur à l'Université de Namur. En tant que Promoteur, il m'a guidé tout au long de la construction de ce travail par ses recommandations, son soutien et son temps consacré.

Aussi, j'aimerais remercier son assistante et co-promoteur, Madame Julie Henry, pour les conseils prodigués et les pistes d'améliorations données.

Il me semble inévitable de remercier mes proches pour leur soutien sans faille lors de ces années d'études. Je souhaitais, plus particulièrement remercier, ma compagne, Céline Thomas, pour son soutien moral, sa disponibilité et son temps passé à la relecture ainsi que ma belle-sœur, Séverine Thomas, pour sa précieuse aide à la relecture et à la correction de mon mémoire.

Chapitre 1

Introduction

De nos jours, les technologies numériques sont partout : de l'utilisation d'un ordinateur ou d'une tablette connectée pour surfer sur internet, à l'achat d'un ticket de train via une borne automatique. Dès son plus jeune âge, l'enfant est donc entouré de technologies numériques. Il est donc normal qu'il s'y intéresse étant donné leur omniprésence.

Mais, utiliser fréquemment une technologie ne garantit pas un usage efficace, ni une compréhension de cette dernière. Un apprentissage reste malgré tout nécessaire.

Ces dernières années, de plus en plus de stages apparaissent, notamment en période de vacances scolaires. Ceux-ci cherchent à initier les jeunes aux nouvelles technologies à travers différentes activités telles que des jeux sur ordinateur ou tablette, des expériences de robotique, ou la découverte à un langage de programmation en particulier. Bien qu'honorables, ces initiatives permettent-elles une réelle assimilation par l'enfant des notions présentées ? Les enfants perçoivent-ils seulement les notions abordées durant les activités proposées ?

En ce qui concerne l'enseignement, la Fédération Wallonie-Bruxelles a signé le pacte pour un Enseignement d'Excellence [33]. Ce dernier propose, entre autres, d'intégrer dès l'enseignement primaire un apprentissage à la logique du numérique. Le terme "numérique" étant fort vaste, aucun cursus n'a encore été mis au point et personne ne peut réellement dire, au moment d'écrire ce mémoire, ce qui sera fait dans ce domaine. Ce qui apparaît comme certain, c'est que pour arriver à mener à bien cet objectif, de nouvelles méthodes d'enseignement vont donc devoir être pensées.

Une des pistes qui pourrait être proposée pour permettre un apprentissage du numérique au niveau primaire serait l'utilisation de dispositifs tangibles. Ce type de dispositif permettrait notamment à l'enfant d'apprendre certaines notions de programmation tout en s'amusant. Ceux-ci nous intéressent tout particulièrement.

A travers cet écrit, les interfaces tangibles seront présentées. Premièrement, deux contextes seront posés. Ensuite, la méthodologie de travail sera expliquée et la question de recherche définie.

La deuxième partie sera composée d'un état de l'art sur les interfaces tangibles, posant les bases d'une bonne compréhension de cette technologie à travers des définitions et des

descriptions d'outils existants.

En troisième et dernière partie, des classifications spécifiques seront présentées permettant de cadrer l'analyse d'une sélection d'outils liés à l'enseignement de la programmation.

Cet ouvrage se termine par une conclusion et des pistes possibles pour continuer la recherche abordée ici.

Chapitre 2

Contexte

Le pacte pour un Enseignement d'Excellence

Face à l'omniprésence de la technologie numérique dans le monde actuel, la Fédération Wallonie Bruxelles a signé, en 2015, le pacte pour un Enseignement d'Excellence qui stipule que « dès l'enseignement primaire, une initiation à la logique du numérique peut utilement être réalisée par la programmation de machines simples » [33].

Jusqu'à présent, aucun cours d'informatique n'existe dans le tronc commun des cours en région Francophone. Ce pacte envisagerait donc d'en introduire un nouveau. Il permettrait d'y aborder certaines notions de "codage". Ceci est une opportunité de réintroduire les concepts de programmations chez les jeunes de cinq à quinze ans.

Les difficultés d'apprentissage des concepts de programmation [8]

L'apprentissage de la programmation est réputé pour sa complexité, mais cela sans que soit apportées de preuves concrètes de cette affirmation. Trois causes seraient envisageables. La première serait liée à la difficulté du contenu à assimiler. Certains chercheurs l'ont représentée par un taux d'échecs fort élevé par rapport aux autres matières (Watson and Li, 2014 ; Bennedsen and Caspersen, 2007). D'autres se basent sur des statistiques collectées via des formulaires de questionnement provenant de la perception, de la compréhension des notions de programmation qu'ont les étudiants (Piteira and Costa, 2012 ; Tan, Ting, and Ling, 2009 ; Lahtinen, Ala-Mutka, and Järvinen, 2005 ; Milne and Rowe, 2002).

La deuxième cause serait à incriminer à l'étudiant : manque de motivation et d'engagement (Alaou-tinen and Smolander, 2010 ; Gomes and Mendes, 2008 ; Kinnunen and Malmi, 2006 ; Jenkins, 2001). Des chercheurs ont voulu dès lors mettre en avant les caractéristiques sociologiques et psychologiques (White and Sivitanides, 2002) des étudiants réussissant les cours de programmation.

Le troisième trouverait, quant à elle, son origine chez l'enseignant : il aurait des attentes trop élevées (Luxton-Reilly, 2016 ; Utting et al., 2013). Cela proviendrait d'un manquement didactique mais également des méthodes d'enseignement ou des outils mal adaptées.

Ce mémoire n'a pas pour objectif de démontrer que la programmation est difficile à apprendre ou de réfléchir à une *Xième* cause expliquant cette difficulté. Ce mémoire va se concentrer sur des outils pensés pour faciliter l'apprentissage de certains concepts de programmation chez les jeunes : les interfaces tangibles.

Chapitre 3

Problématique et méthodologie de recherche

Ce mémoire, en plus de présenter différents outils existant dans le domaine des interfaces tangibles utilisateur pour l'apprentissage de la programmation, va tenter d'apporter des éléments de réponse à la question suivante :

« Les interfaces tangibles utilisateur permettent-elles aux jeunes de comprendre les concepts de base en programmation ? »

Pour trouver réponse, cette question a été déclinée en sous-questions plus précises permettant une réponse en étapes :

- Quels concepts de base en programmation, une interface tangible utilisateur permet-elle d'aborder ?
- Quelles sont les modalités d'interaction privilégiées entre l'utilisateur et l'interface tangible ?
- Quels sont les métaphores utilisées pour représenter les inputs dans les différents dispositifs tangibles ?

Des recherches ont d'abord été menées sur les interfaces tangibles utilisateur et leurs différentes utilisations afin de se familiariser avec le sujet principal du mémoire. Le cours « Advanced Human Machine Interaction : 6 - Tangible User Interface », dispensé par M. Dumas, a constitué une solide introduction à la thématique.

Des recherches via le moteur de recherche *Google Scholar*¹ ont ensuite permis de collecter des expériences et des articles décrivant des dispositifs tangibles. Ces recherches ont été menées, dans un premier temps, avec les mots-clés suivants : "Tangible User Interface", "Learning", "Children". Grâce à celles-ci, des articles intéressants parlant de l'utilisation des interfaces tangibles utilisateur pour l'apprentissage chez les jeunes, ont été trouvés.

Pour affiner ces résultats, les mots "Programming Tool", "Programmation" et "School", ont été ajoutés. Cette recherche plus précise a permis de mettre en avant une dizaine

1. <https://scholar.google.be/>

d'articles parlant de l'utilisation des interfaces tangibles utilisateur pour l'apprentissage de la programmation chez les jeunes.

A partir de ces articles et grâce aux références d'expériences trouvées dans ceux-ci, des nouveaux liens ont été faits vers d'autres articles relatant ce sujet. Cette opération a été aussi réalisée avec le nom des auteurs.

Au final, une vingtaine d'articles discutant de dispositifs tangibles permettant l'apprentissage de concepts de programmation chez les jeunes ont été trouvés.

Suite à la lecture de chaque expérience ou article trouvés, les articles hors sujet ont été éliminés. Après ce tri, les articles correspondant au sujet de la recherche ont, quant à eux, été résumés en fiches descriptives afin de garder une vue globale de chaque dispositif.

C'est sur base de ces vingt dispositifs que la prochaine étape du mémoire a été rédigée. Il s'agit d'une partie de recherche et de classification qui allait permettre d'apporter les réponses aux 3 sous-questions.

Quels concepts de base en programmation une interface tangible utilisateur permet-elle d'aborder ?

Pour répondre à cette question, un travail de recherche sur les concepts de base en programmation a été effectué afin de les recenser et de les expliquer un à un. Ensuite, une classification des différents outils a été faite permettant d'avoir une vue d'ensemble sur les concepts abordés lors de l'utilisation d'une interface tangible utilisateur.

Quelles sont les méthodes d'interaction privilégiées entre l'utilisateur et l'interface tangible ?

L'approche de travail pour répondre à cette question ressemble à celle de la question précédente. Ce qui les différencie, c'est la recherche des méthodes d'interactions entre l'utilisateur et l'outil qui a été faite en concertation avec Monsieur Dumas et Madame Henry sur base de l'expertise de chacun. Lorsqu'une classification correcte a été trouvée, différents tableaux reprenant les critères ont été construits dans le but d'avoir une vue globale des différentes méthodes d'interaction entre l'utilisateur et les dispositifs tangibles. Une des parties du tableau permettant la classification des différents types d'interaction entre l'utilisateur et les dispositifs se base sur une classification mise au point par Roudaut. [19]

Quels sont les types de métaphores utilisés pour représenter les inputs dans les différents dispositifs tangibles selon Fishkin ?

Afin de répondre à cette dernière question, les différents dispositifs tangibles recensés ont été classés selon une taxonomie réputée dans le domaine des interfaces tangibles : la taxonomie de Fishkin [4].

En plus de cela, dans le but d'enrichir les observations et les analyses, une classification propre a été mise en place. Après ce long travail de classement des différents dispositifs,

des observations et des analyses ont été effectuées sur base des différents critères afin d'essayer de répondre aux trois sous-questions.

Chapitre 4

Etat de l'art

4.1 Interface tangible utilisateur

4.1.1 Définition, utilisation et illustration

Selon des professionnels reconnus dans le domaine (Sophie Lepreux, Julien Castet, Nadinne Coutue, Emmanuel Dubois, Chistophe Kolski, Sébastien Kubicki, Valérie Maquil, Guillaume Riviere) [16], une interface tangible utilisateur est une : « Interface homme-machine permettant aux utilisateurs d'interagir avec un système à l'aide d'outils, d'objets physiques afin d'en dégager certains résultats. »

Décortiquée mot à mot, l'expression « interface tangible utilisateur », se précise grâce au petit Robert en ligne :

1. Interface : « En informatique, jonction entre deux matériels ou logiciels leur permettant d'échanger des informations par l'adoption de règles communes ; module, matériel ou logiciel permettant la communication d'un système avec l'extérieur. »
2. Tangible : « qu'on connaît par le toucher ; matériel, sensible »
3. Utilisateur : « Personne, groupe qui fait usage de quelque chose, qui utilise un appareil, un service. »

Partant des définitions, une interface tangible utilisateur est : « une interface avec laquelle une personne doit utiliser des objets physiques dans le but d'établir une communication entre lui et le système informatique de celle-ci afin d'obtenir un résultat ».

Afin d'éviter la répétition des termes « interface tangible utilisateur », au cours de ce mémoire, l'utilisation de l'abréviation TUI sera employée. Celle-ci provient de l'anglais « tangible user interface » citée pour la première fois par Ishii [31].

La présentation du modèle d'interaction (figure 1) entre l'utilisateur et la TUI permettra une meilleure compréhension de l'utilisation de cette dernière.

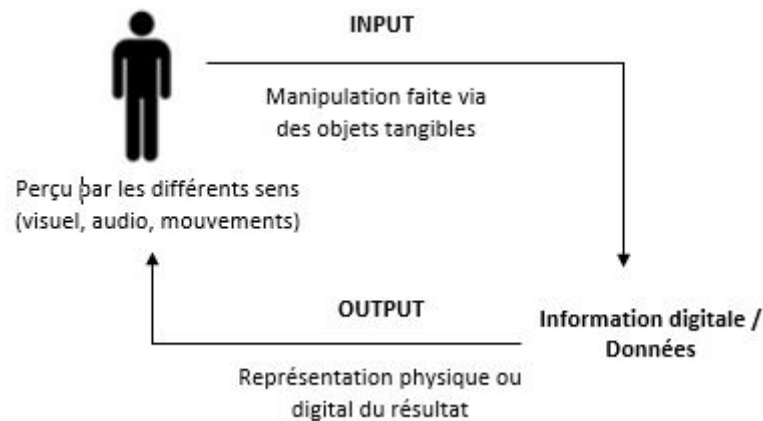


FIGURE 1 – Modèle d’interaction

Dans un premier temps, l’utilisateur de l’interface va communiquer avec le système par le biais d’objets physiques, les INPUTS afin de lui poser une question ou de lui transmettre des informations. Le système, après avoir interprété les informations reçues, va quant à lui, répondre à la demande via l’OUTPUT. Le résultat sera perçu par l’utilisateur à travers ses différents sens, u compris de manière tangible.

Pour compléter l’explication, une présentation de différents TUI est proposée afin de visualiser quelques illustrations concrètes : CurlyBot , MusicBottles, URP, Audiopad, Sandscape, I/O Brush.

CurlyBot (Figure 2) est un robot qui reproduit identiquement les déplacements qu’on lui fait faire. Comme le montre la vidéo ¹, l’utilisateur appuie le bouton central du robot, déplace le robot qui enregistre les déplacements et termine l’enregistrement en ré appuyant sur le bouton. Le robot se met ensuite en action en reproduisant le déplacement. Dans le cas de cet outil, l’input et l’output ne sont autre que le robot même.



1. <http://tangible.media.mit.edu/project/curlybot/>

FIGURE 2 – CurlyBot

MusicBottle (Figure 3) est une table interactive² qui réagit en musique lorsque l'utilisateur déplace, dé bouchonne, change de bouteille. Chaque bouteille représente un instrument différent. Dans ce cas-ci, les inputs sont les bouteilles que l'utilisateur manipule et les outputs sont la musique émise par la table et les variations de couleurs affichées par celle-ci.



FIGURE 3 – MusicBottle

URP (Figure 4) est un outil de planification urbain qui permet de prendre en considération certains paramètres comme le vent, les ombres etc. Dans la vidéo³, l'utilisateur place sur la table les structures de bâtiments et en activant le TUI, celui-ci va calculer et afficher les résultats des ombres que provoquent le bâtiment, ou les trajectoires du vent selon la demande de l'utilisateur. Pour cet outil, les inputs sont les structures de bâtiments qui sont posées sur la table interaction ainsi que les différents critères que l'utilisateur définit. En ce qui concerne les outputs, celui-ci est l'affichage du résultat sur la table interactive.

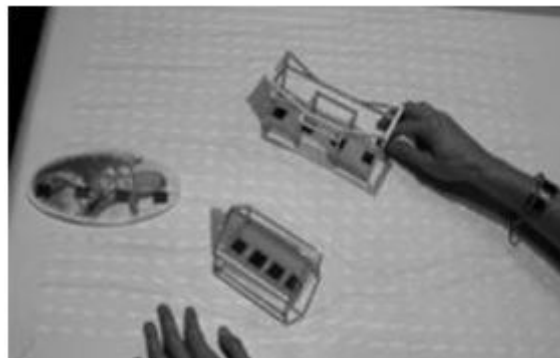


FIGURE 4 – URP

2. <http://tangible.media.mit.edu/project/musicbottles/>

3. <https://vimeo.com/48600713>

AudioPad⁴ (Figure 5) permet la composition de musiques via des interactions d'objets physiques sur une table interactive. Chaque mouvement des inputs permet d'ajouter, de modifier, de retirer un effet de musique. Ce TUI ressemble à une table de mixage mais plus interactive. Les inputs de cet outil sont les objets physiques que l'utilisateur pose sur la table afin de mixer la musique tandis que les outputs sont la musique générée et les effets visuels visibles sur la table interactive.



FIGURE 5 – AudioPad

Sandscape⁵ (Figure 6) est un outil qui permet certaines analyses (l'inclinaison, la hauteur, les ombres, l'écoulement d'eau) d'un paysage modelé par l'utilisateur dans du sable. Pour cet outil, les inputs sont le bloc permettant de choisir l'analyse à effectuer et le sable qui est sur la table. L'output est, quant à lui, l'effet visuel rendu sur le sable.

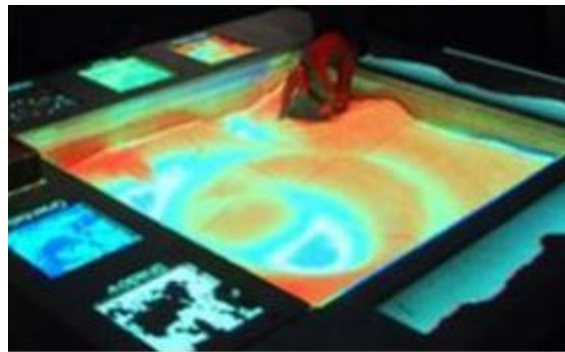


FIGURE 6 – Sandscape

I/O Brush (Figure 7) est un dispositif⁶ composé d'un pinceau qui permet de reproduire la couleur ou l'image que celui-ci a capturé sur un écran. Pour cette interface tangible, l'input est le pinceau que l'on applique sur les couleurs ou images à reproduire. L'output est, quant à lui, le rendu visuel qui est affiché sur l'écran.

4. <https://www.youtube.com/watch?v=59zS7m23IsM>

5. <https://www.youtube.com/watch?v=XpyJcN3Tyzg>

6. https://www.youtube.com/watch?v=04v_v1gnyO8



FIGURE 7 – MusicBottle

Tous ces exemples permettent de constater qu'il existe une multitude de TUI différentes tant au niveau des types d'interfaces, qu'au niveau des différents inputs. Plusieurs taxonomies permettent de classifier toutes ces TUI.

4.1.2 Taxonomies

Avant de présenter différentes taxonomies, il est intéressant de préciser qu'il s'agit de méthodes de classification selon plusieurs critères.

Dans le monde des TUI, il existe plusieurs taxonomies permettant le classement des différents dispositifs.

Une des premières est la Taxonomie de Ullmer et Ishii qui a été mise en place en 2000 [17]. Celle-ci classe les dispositifs tangibles en 4 groupes de système : système spatial, système ayant une approche constructive, système rationnel et système associatif.

Une autre taxonomie est celle de Van den Hoven et Eggen apparue en 2004[17]. Cette taxonomie classe les dispositifs selon que les éléments physiques ont une signification personnelle pour l'utilisateur ou non. Des sous-groupes sont ensuite établis sur le fait que les associations sont flexibles ou fixes et iconiques ou symboliques.

La taxonomie la plus intéressante pour notre recherche est celle de Fishkin créée en 2004 [4]. Cette taxonomie se base sur deux grands axes : l'incarnation qui permet de définir la distinction qu'il y a entre l'INPUT et l'OUTPUT d'une TUI et la métaphore qui juge la relation entre l'action de l'utilisateur sur le TUI et la même action dans le monde réel. Ces deux axes sont fractionnés en 4 sous-groupes.

L'incarnation est soit :

- complète, ce qui signifie que l'output est l'input. le résultat des actions commandées par les inputs sont visibles sur les inputs mêmes.
- Proche, ce qui signifie que le résultat se produit sur un élément proche de l'input.
- Environnementale, ce qui signifie que l'output incarne généralement un ou plusieurs effets : effet sonore, effet de lumière... Le résultat se produit autour de l'utilisateur.
- Distant, ce qui signifie que le résultat est affiché sur un élément éloigné de l'utilisateur.

Concernant l'axe métaphore, les sous-groupes sont :

- nom, lorsque la représentation de l'input ressemble à son apparence dans le monde réel tant au niveau visuel qu'au niveau auditif.
- verbe, lorsqu'il existe une analogie entre l'action réalisée par l'input indépendamment de l'objet qui amène l'action.
- nom & verbe, lorsque la représentation de l'input regroupe les deux types précédents
- complète, ce qui signifie qu'aucune analogie ne doit être faite. Le système virtuel est le système physique.

4.1.3 Les TUI et l'apprentissage chez les jeunes

Selon « la théorie de Piaget » [7, 3], les enfants passent par plusieurs stades de développement intellectuel : le stade Sensorimoteur (de 0 à 2 ans), le stade des préopérateur de (2 à 7 ans), le stade des opérations concrètes de (7 à 12 ans) et le stade des opérations formelles de (12 à 19 ans).

Ce mémoire traitant de l'utilisation d'outils pour l'apprentissage de concepts de programmation chez les enfants des écoles maternelles et primaires, nous nous limiterons ici aux stades préopérateur et des opérations concrètes (de 2 à 12 ans).

Le stade préopérateur

Dès 3 ans, avec l'entrée de l'enfant à l'école, l'enfant franchit une étape importante de sa vie : la scolarisation. Celle-ci va entraîner le début de la sociabilisation avec les autres enfants, le début de l'apprentissage de la communication et la soif de découverte.

Le stade des opérations concrètes

Au cours de ce stade, l'enfant va commencer à penser de manière logique. Il sera donc capable de réaliser des tâches plus complexes tel que des mathématiques. Mais attention, cette logique ne s'applique qu'à l'ici et maintenant, il leur est encore très difficile de penser de manière abstraite.

Les traits caractéristiques du développement chez les enfants pour ces deux groupes d'âge sont d'ailleurs fortement repris dans les résultats des différentes expériences menées par les chercheurs. Beaucoup d'expériences ont démontré que, grâce aux TUI, les enfants augmentaient leur travail en collaboration et les échanges constructifs [1, 23], ou que ceux-ci prenaient du plaisir à découvrir les nouveaux dispositifs [23, 41].

Un autre point montrant que la théorie de Piaget s'accorde bien avec l'utilisation des TUI au niveau de l'apprentissage de concepts de programmation est que régulièrement les plus jeunes éprouvent des difficultés à utiliser certains concepts plus compliqués comme la boucle [30, 35].

En plus de l'utilisation de GUI (Graphical User Interface : est un type d'interface permettant à une personne de communiquer avec une machine au moyen de symboles, de métaphores visuelles et de dispositifs de pointage) en milieu scolaire, il n'est pas anodin que les TUI soient de plus en plus testées et utilisées pour l'apprentissage chez les jeunes. De plus, il a été prouvé [39] que l'utilisation de TUI au niveau scolaire permet aux jeunes apprenants de pouvoir mettre une image sur les actions faites pour la réalisation d'une tâche.

Voici deux exemples illustrant cet outil dans l'apprentissage :

Le premier permet la découverte d'une culture avec la mise en place, au cours de l'exercice, d'un album photo permettant de garder un souvenir « du voyage ». Grâce à la confection de cet album, l'utilisateur garde en tête ce qu'il a appris en accrochant une image à des mots ou des actions. Le deuxième, quant à lui, est la lecture d'un livre

interactif en forme de cube avec de la réalité augmentée permettant une plus grande immersion de l'enfant.

Designing Tangible Interfaces for Children's collaboration [1]

Cette expérience d'utilisation de TUI ne rentre pas dans le domaine de l'apprentissage de concepts de programmation. Néanmoins, il montre que l'utilisation d'une interface tangible utilisateur chez les jeunes en milieu scolaire peut augmenter leur concentration et leur collaboration.

Selon cet article, il s'agit d'une TUI « Ely the explorer » qui permet l'apprentissage d'une culture étrangère aux enfants. Cette TUI permet la collaboration entre élèves et l'application a été développée pour le milieu scolaire.

D'un point de vue matérielle, cette TUI est composée de :

- une table interactive avec écran tactile ;
- un ensemble d'outils : Bouton Rotary, RFID, Cartes, Lecteur cartes RFID, PDA (Assistance digital personnel), caméra digitale.

« Ely » est l'agent qui guide les enfants durant le jeu. Il donnera les instructions et les résultats d'interactions. Au cours du jeu, les enfants devront interagir avec la table interactive en résolvant des problèmes (mathématique, de lecture, d'écriture ...) pour avancer dans l'histoire et découvrir de plus en plus de la nouvelle culture. Cette table interactive est scindée en 3 parties individuelles et une partie commune. Les enfants collaborent donc les uns avec les autres. (figure 8)

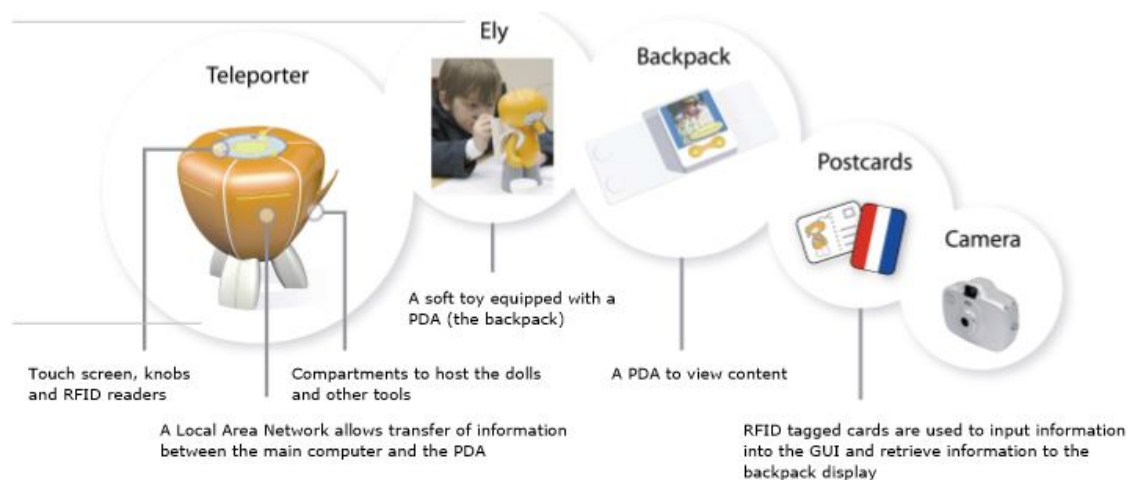


FIGURE 8 – Présentation du TUI *Ely the explorer*

Les résultats de ce test en milieu scolaire (enfants de sept-huit ans) montrent que l'utilisation de cette TUI pousse les enfants à la collaboration et que ces derniers sont concentrés, engagés dans la résolution du problème. Au niveau du corps professoral, les retours sont aussi positifs car l'utilisation de ce genre de TUI est motivant et propice à l'introduction de nouvelles technologies dans le milieu scolaire.

Magic story cube : an interactive tangible interface for storytelling [41]

Cette expérience d'utilisation de TUI ne rentre pas dans le domaine de l'apprentissage de concepts de programmation mais introduit un concept qui pourrait être utilisé dans une expérience future permettant une plus grande immersion : la réalité augmentée

Il s'agit d'une TUI qui souhaite améliorer le livre pour enfant en modifiant l'aspect de celui-ci (forme cubique) et en y ajoutant une plus grande immersion grâce à l'utilisation de la réalité augmentée. (figure 9)

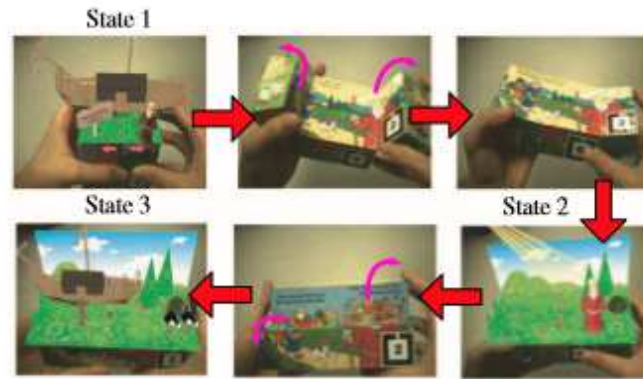


FIGURE 9 – Présentation du TUI *Magic story cube*

L'enfant doit être équipé de lunettes permettant l'utilisation de la réalité augmentée. Il doit manipuler le cube dans un ordre particulier pour que l'histoire suive son cours logique.

Les résultats de cette expérience montrent que l'histoire sous cette forme est plus attractive et plus compréhensible grâce à la réalité augmentée. Mais d'autres expériences doivent être menées pour appuyer ces dires.

4.2 Les concepts de base de la programmation

La programmation est une matière complexe à assimiler, à comprendre et à maîtriser. Lorsqu'il s'agit d'enfants, les concepts de base nécessaires à une bonne initiation doivent être définis. Ils peuvent s'inspirer des concepts dispensés dans un cours d'introduction à la programmation, par exemple celui de M. Benoit Frénay [32] à l'Université de Namur.

Algorithme / Programme

Un algorithme correspond généralement à la description systématique d'un processus, sous forme d'une série d'instructions dont l'exécution aboutit au résultat souhaité. Il faut noter que la notion d'algorithme n'est pas propre au domaine de l'informatique. Dans le contexte informatique, je me limiterai aux algorithmes du genre traitement de données. Dans ce cas, l'algorithme est considéré comme une série d'instructions qui permet de transformer un ensemble de données initiales (des informations représentées sous une forme quelconque) en résultats. [32]

Séquence

Une séquence est une suite ordonnée d'opérations et d'éléments à traiter.

Structure conditionnelle

Dans le jargon informatique, la structure conditionnelle est aussi appelée « structure SI (condition) ALORS (séquence d'instruction) ». Il existe plusieurs types de structures conditionnelles :

- SI (condition respectée) ALORS exécution de la séquence d'instructions ;
- SI (condition respectée) ALORS exécution de la séquence d'instructions SINON exécution d'une autre séquence d'instructions.

Cette structure permet de proposer des traitements différents si une condition est respectée ou non.

Boucle

La boucle permet la répétition d'une séquence d'actions tant qu'une condition de sortie n'est pas atteinte. Il en existe plusieurs :

- boucle « tant que » : tant que la condition n'est pas atteinte, il faut exécuter la séquence d'instructions dans la boucle ;
- boucle « jusqu'à ce que » : répéter la séquence d'instructions JUSQU'A CE QUE la condition soit atteinte ;
- l'itérateur : méthode permettant de parcourir tous les éléments d'une structure de données.

Variable

La variable est un concept fondamental dans la programmation informatique. Il s'agit de l'association d'un nom à une valeur. Dans l'utilisation des TUI, la variable sera un input auquel un nombre est associé. Par exemple, le nombre de fois qu'une boucle doit être reproduite ou alors le nombre de degrés de rotation avec lequel un changement de direction doit être opéré lors de la manipulation d'un robot. Elle permettra de définir un paramètre pour une action.

Sous-programme : appel de fonction / Procédure

Dans le domaine de la TUI, le sous-programme sera une séquence d'actions, préalablement encodée qui pourra être appelée lors de la programmation de la séquence principale.

Chapitre 5

Contributions personnelles

Dispositifs tangibles sélectionnés

Cette partie du mémoire va lister les vingt dispositifs sélectionnés lors de la recherche :

Nom des outils	
1) Algoblock	11) StarLoop
2) Bee-Bot	12) Strawbies
3) Dialando	13) Tangicons
4) Dr. Wagon	14) Tan-Pro Kit
5) E-Block	15) T-Butterfly
6) Electronic Blocks	16) Tern
7) KIBO	17) T-Maze
8) Pleo	18) Toque
9) Quetzal	19) T-ProRob
10) Robo-Blocks	20) TurTan

Description des dispositifs

Cette partie de mémoire va référencer, sous forme de fiches descriptives, les différents outils créés par des chercheurs dans le but d'apprendre les concepts de programmation aux enfants de moins de douze ans.



Chaque fiche descriptive sera composée d'informations générales relatives au dispositif tel que le nom, une brève description, l'âge requis pour l'utilisation, ainsi qu'une illustration du dispositif. Toutes ces informations permettent de se faire une représentation globale de la TUI.

Ensuite cette partie proposera une description plus technique de l'outil. Elle présentera les technologies utilisées à la conception du dispositif, une description des inputs et outputs ainsi que la méthode de programmation que l'utilisateur doit appliquer pour le bon fonctionnement de la TUI.

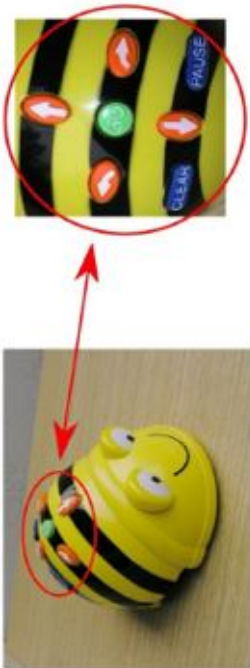
Pour terminer, les fiches présenteront les études menées, la population touchée et les résultats obtenus par les chercheurs lors de leurs expérimentations ainsi que les améliorations futures pensées pour leurs dispositifs.


Fiches descriptives

AlgoBlock [28]


AlgoBlock	
Description générale de la TUI	<p>Pour utiliser cette TUI, l'utilisateur doit connecter des cubes physiques les uns aux autres. Ces cubes connectés formeront la séquence d'actions que le programme devra exécuter par un ordinateur pour faire bouger une fusée.</p> <div>   </div>
Input / Output	<p>Input : blocs qui se connectent entre eux. Il en existe de plusieurs types : des blocs permettant de faire avancer, tourner, décoller le véhicule et d'autres blocs exécutant des conditions (IF/ELSE) ou des répétitions (Boucle). Cette TUI propose aussi l'utilisation de paramètres. Chaque action est représentée de manière illustrée sur les cubes.</p> <p>Output : Le résultat de la séquence des blocs est visible sur l'écran d'ordinateur après que le programme ait interprété celle-ci.</p>
Technologie	<p>AlgoBlock : Un langage de programmation tangible de la famille LOGO</p> <p>L'article ne parle pas des technologies utilisées dans les Inputs et des différentes connectiques dans et entre les cubes.</p>
Méthode de programmation	<p>Afin de programmer la séquence, l'utilisateur doit connecter les cubes ensemble. Chaque cube représente une action que le véhicule doit effectuer. Lorsque la séquence est prête, l'utilisateur l'exécute pour vérifier le résultat.</p>
Tranche d'âge	12 ans
Etudes de l'expérience	<ol style="list-style-type: none"> 1) La visualisation des effets de l'AlgoBlock sur le travail en collaboration. 2) L'augmentation des compétences des jeunes utilisateurs à construire et planifier des procédures. 3) L'Apprentissage de certains concepts de programmation : la planification des séquences, l'exécution de la séquence, l'utilisation des conditions et des boucles, l'utilisation de paramètres.
Population touchée	3 enfants (filles et garçons)
Résultats obtenus	<ul style="list-style-type: none"> - Les modifications du programme se font avec facilité après l'exécution de la séquence. - La prise en main du langage est facile. Il aura fallu après la phase d'introduction 3 minutes aux enfants pour comprendre le langage.
Futur	Non défini


Bee-Bot [15]


Bee-Bot	
Description de la TUI	<p>Le robot se nomme Bee-Bot. Il se présente sous la forme d'une abeille avec 7 boutons sur son dos : GO / CLEAR / PAUSE / AVANCER / RECULER / TOURNER DROITE / TOURNER GAUCHE. Ces actions permettent aux utilisateurs de la TUI d'introduire les mouvements que le robot doit reproduire.</p> 
Technologie	Le robot possède une technologie embarquée capable de garder en mémoire une séquence d'actions à réaliser.
Input / Output	L'input est l'output. Il s'agit d'un robot en forme « d'abeille » de couleur jaune et noir avec des boutons de commande rouges et verts.
Méthode de programmation	La séquence de programmation correspond aux actions que l'utilisateur encode dans la mémoire de l'abeille grâce aux boutons de commande. Lorsque la séquence est programmée, l'utilisateur doit l'exécuter.
Tranche d'âge	De 4 à 6 ans
Etudes effectuées	<ol style="list-style-type: none"> 1) Familiarisation avec les commandes de manipulation du robot. 2) Utilisation des commandes de manière séquentielle ou en pas à pas (par essai - erreur). 3) Programmation des actions du jouet pour effectuer des trajets prédéfinis.
Population touchée	7 classes d'écoles maternelles, plus ou moins 100 élèves
Résultats obtenus	<ul style="list-style-type: none"> - La possibilité d'apprentissage de certains concepts de programmation grâce à une scénarisation pédagogique adéquate. - La facilité de compréhension des actions représentées sur la TUI : compréhension plus aisée des touches directionnelles que des boutons GO/ CLEAR / PAUSE et que la notion de mise en mémoire. - L'apprentissage de la conception d'un premier programme tant sur le plan syntaxique que sémantique : remise de la mémoire à vide, introduction des commandes et exécution du programme (programmation séquentielle pas à pas suivi d'une programmation d'une séquence complète). - Une meilleure compréhension de l'enfant et l'optimisation de sa motivation passent inévitablement par une bonne scénarisation. Les enfants sont capables, en général, de construire des programmes séquentiels sur la base de commandes visuelles (cartes) et de les transposer vers le jouet.
Futur	Non défini dans l'article scientifique


Dialando	
Description de la TUI	Mise au point d'une TUI bon marché qui utilise des matériaux recyclés dans le but d'introduire des notions de concepts de programmation aux utilisateurs. Pour ce faire, l'utilisateur devra programmer une séquence d'actions composée de flèches directionnelles afin de faire bouger un objet d'un point A vers un point B.
	
Technologie	Le système est composé de boîtes sur lesquelles sont posés des boutons directionnels rotatifs permettant d'indiquer les directions de déplacement que l'objet doit effectuer. Il comprend également une boîte servant de connexion entre le système et les inputs via USB. Le résultat est affiché sur un écran d'ordinateur.
Input	Input : boîtes sur lesquelles des boutons rotatifs sont placés. Une représentation d'une flèche directionnelle est illustrée sur chaque bouton. Cela permet à l'utilisateur de générer sa séquence d'actions en définissant les directions que doit prendre l'objet. Output : écran d'ordinateur sur lequel le résultat est affiché.
Méthode de programmation	Afin de programmer la séquence que l'objet doit exécuter, l'utilisateur doit aligner des boutons rotatifs sur lesquels les directions que l'objet doit suivre sont représentées. Cette séquence est composée de maximum 5 directions consécutives.
Tranche d'âge	Non défini
Etudes effectuées	Mettre au point une TUI qui soit bon marché afin de pouvoir apprendre aux jeunes des pays en voie de développement.
Population touchée	Enfants venant d'un milieu en voie de développement.
Résultats obtenus	Aucun test n'a été fait.
Futur	/


Dr. Wagon	
Description du la TUI	<p>Cette TUI permet aux utilisateurs de créer un robot et de programmer ses réactions et ses mouvements. Pour ce faire, l'utilisateur doit former les instructions en jouant avec des blocs de programmation qu'il connecte les uns aux autres.</p> 
Technologie	<p>Les blocs de programmation sont embarqués d'un système électronique composé d'un PIC microcontrôleur à ID unique. Ces blocs se connectent les uns aux autres afin de générer une séquence de programmation. Le bloc Start, via une technologie sans fil, permet la communication entre le robot et la séquence générée par l'utilisateur. Lorsque l'utilisateur appuie sur le bouton de compilation, chaque bloc transmet son ID au bloc Start qui génère une suite avec ces ID et le transmet au robot qui va pouvoir connaître les actions à réaliser.</p> <p>Input : blocs faits en matière « stretch » qui s'emboîtent les uns aux autres et se connectent via 4 contacts magnétiques. Il existe différents types de blocs : Star & End, bloc de mouvement, bloc de condition d'état et boucle.</p> <p>Output : robot qui va réagir avec son environnement grâce à différents capteurs et senseurs (distance, couleur, déclencheurs...)</p>
Méthode de programmation	<p>L'utilisateur doit former des séquences en emboîtant les blocs de programmation, les uns aux autres, pour programmer les réactions du robot.</p>
Tranche d'âge	Les enfants de 6 à 12 ans.
Etudes effectuées	Aucune étude n'a été effectuée dans cet article. Celui-ci décrit seulement un outil permettant l'apprentissage de concepts de programmation.
Population touchée	/
Résultats obtenus	/
Futur	<p>Tester la TUI chez des enfants de 6 à 12 ans et observer si cette dernière remplit les objectifs fixés par les créateurs.</p> <p>Améliorer la connexion entre les différents blocs de programmation.</p> <p>Proposer un plus large catalogue d'actions pour les blocs de programmation.</p>


E-block	
Description de la TUI	<p>E-block est une version améliorée de T-Maze. Elle corrige quelques problèmes présents dans T-maze. Le fond de la TUI reste le même, les améliorations se situent au niveau des technologies utilisées : problème d'occlusion de caméra, etc.</p> <p>Cette TUI est un jeu de labyrinthe. Le but est d'en faire sortir un avatar grâce à des séquences programmées par des cubes et des senseurs.</p> <p>Après la réalisation de la séquence à appliquer, l'utilisateur verra son personnage évoluer sur l'écran d'ordinateur. L'utilisateur a également la possibilité de créer ses propres labyrinthes.</p>
 <p>Figure 3: E-Block tool.</p>	
Input / Output	<p>Input : blocs embarqués d'un microprocesseur et d'un système de communication. Ces blocs sont identiques à ceux du T-Maze. Chaque action est représentée de manière illustrée sur les cubes.</p> <p>Output : un autre changement se situe au niveau de l'output, un système de LED intégré aux blocs informera l'utilisateur d'une erreur dans sa séquence.</p>
Technologie	<p>La différence avec T-maze se situe à ce niveau : les blocs d'input sont composés d'une puce microprocesseur, d'un transmetteur et d'un récepteur infrarouge permettant la communication entre blocs, d'un module sans fil permettant la communication avec le système informatique et d'un module LED avertissant l'utilisateur d'erreurs possibles dans la séquence. La caméra permettant de lire la séquence générée par l'utilisateur n'existe plus.</p>
Méthode de programmation	<p>Comme pour les styles précédents, l'utilisateur programme la séquence d'actions en alignant des cubes d'action. Chaque cube placé est une action qui va être réalisée. Dans ce cas, la caméra a été remplacée par des composants de communication sans fil.</p>
Tranche d'âge	De 5 à 9 ans
But de l'expérience	L'expérience a pour but de comparer l'utilisabilité entre le T-maze, la caméra et le nouveau système E-block.
Population touchée	11 enfants (3 garçons / 8 filles).
Résultats obtenus	<ul style="list-style-type: none">- Rapide présentation sur pc en temps réel ;- moins de problèmes de place au niveau des blocs ;- moins de problèmes d'occlusion de la caméra ;- le feedback apparaît sur l'écran et les blocs.
Futur	<ul style="list-style-type: none">- Introduire davantage de blocs programmés et d'autres scénarii.- Essayer de ne plus passer par un OUTPUT sur écran d'ordinateur mais uniquement sur les blocs.



Electronic blocks	
Description de la TUI	<p>Il s'agit de blocs électroniques que l'enfant empile les uns sur les autres pour former une structure pouvant interagir avec son environnement. Cette construction forme un « programme informatique ». Chaque bloc permet une fonction différente.</p> 
Technologie	Des blocs en plastique avec un circuit électronique embarqué (placé à l'intérieur de Lego, Duplo Primo blocks)
Input / Output	Il s'agit de blocs embarqués de différentes couleurs empilés les uns sur les autres (I/O). Chaque bloc a une fonction en particulier (Sensor / Action / Logic).
Méthode de programmation	La programmation est définie par les blocs empilés les uns sur les autres (de manière verticale). Chaque bloc ajoute une fonctionnalité au « Robot », une action à faire ou une manière de réagir face à un événement.
Tranche d'âge	De 4 ans à 8 ans
Etudes effectuées	<ol style="list-style-type: none"> 1) L'apprentissage des concepts de programmation : compréhension de la syntaxe, encodage de structure simple, débog des empilements de blocs. 2) L'attitude de l'enfant face à la TUI.
Population touchée	40 enfants (12 sessions expérimentales)
Résultats obtenus	<ul style="list-style-type: none"> - Le niveau d'attention des enfants : très élevé avec un grand intérêt (préscolaire : environ 30 minutes). - La compréhension des fonctionnalités des blocs : elle varie selon l'âge des utilisateurs. Plus ils sont âgés, plus cela est aisé. - Le niveau de frustration : il dépend de la réussite de l'objectif ou non. - La compréhension de la syntaxe : étant donné que ce sont des blocs à empiler les enfants comprennent facilement. - La programmation d'une structure simple : les enfants plus âgés ont quasiment tous réussi à obtenir le résultat souhaité. En revanche, des difficultés d'utilisation et de compréhension sont observées chez les enfants plus jeunes. - Le debugging programme : les enfants plus âgés ont davantage de facilités à l'appliquer. De temps en temps, les plus jeunes n'ont pas assez d'informations pour le comprendre. - La réutilisation des fragments de code : les enfants ont réutilisé les exemples de démonstration pour commencer leur séquence.
Futur	Améliorer le design, augmenter la panoplie des senseurs, ajouter des blocs différents (condition logique etc...), développer plus de jeux et de scénarii.


Kibo	
Description de la TUI	<p>Il s'agit d'un robot prénommé KIBO. Le but consiste à laisser l'enfant construire son propre robot en utilisant des moteurs, des senseurs et des matériaux de construction pour, ensuite, lui faire réaliser des actions grâce à la formation de séquences d'actions avec des cubes.</p> 
Technologie	<p>Il s'agit d'un robot que l'enfant assemble lui-même. Ce robot est donc composé d'une base sur laquelle l'enfant va placer différents composants pour le faire évoluer selon son envie (bloc décoratif, senseur, moteurs...). Le robot est aussi composé d'une caméra permettant la lecture des codes actions afin de réaliser les séquences demandées.</p>
Input / output	<p>Input : les cubes en bois sur lesquels l'action et un code barre sont dessinés afin de construire la séquence d'actions que le robot doit réaliser. Les cubes proposent les actions suivantes : les déplacements du robot, la réalisation d'une boucle, et l'utilisation de variables.</p> <p>Output : le robot. Selon les composants que l'enfant a placés dessus, le robot va émettre des sons, de la lumière ou simplement exécuter les déplacements programmés.</p>
Méthode de programmation	Le robot scanne les codes-barres présents sur les cubes en bois. Cela lui permettra de programmer la séquence d'actions qu'il devra réaliser.
Tranche d'âge	De 4 ans à 7 ans
Etudes effectuées	Au cours de ces expériences avec les enfants et les professeurs, les chercheurs ont surtout regardé à améliorer l'aspect design et l'utilisabilité du robot.
Population touchée	Jeunes enfants en camp ou à l'école. Nombre inconnu.
Résultats obtenus	<ul style="list-style-type: none"> - La prise en main de l'outil s'accroît. - Les enfants peuvent apprendre les concepts de programmation. L'outil permet l'intégration de compétences techniques en relation avec les maths, la littérature, etc. - Kibo permet de développer le travail en collaboration.
Futur	Tester la dernière version du KIBO afin d'observer si des évolutions et/ou des améliorations en terme de design doivent encore être réalisées.


Pleo	
Description générale de la TUI	<p>Cette TUI permet la création d'histoires avec un dinosaure comme personnage principal. Celui-ci prend vie en réagissant aux différentes cartes que l'utilisateur lui montre soit lorsque l'utilisateur touche un une ou l'autre partie de son corps. Toutes les réactions du dinosaure sont programmées par l'utilisateur via une tablette.</p> 
Input / Output	<p>Input : 12 cartes, munies d'un bord de couleurs et d'un code unique, que les enfants illustrent d'un dessin. Les autres inputs sont les parties du corps du dinosaure que l'enfant peut toucher. Les actions qui sont liées à chaque partie du corps ou aux dessins réalisés sont préprogrammées mais peuvent être modifiées.</p> <p>Output : dinosaure PLEO connecté via USB à l'ordinateur. C'est à travers ses réactions que le résultat sera visible.</p>
Technologie	<p>La TUI est composée d'une table interactive (118 x 78 x 49 cm) sur laquelle sont posés : un dinosaure (output), une tablette PC (12 pouces) et des cartes à dessiner.</p> <p>La table interactive permet de lire, via une caméra, les cartes dessinées par l'enfant grâce au bord coloré et au code unique. Le dinosaure est embarqué de capteurs différents : capteurs sonores permettant l'enregistrement du son et capteurs permettant de détecter les pressions exercées par l'enfant lorsqu'il touche une partie du corps. La tablette permet de programmer la réaction que le dinosaure a, face à une interaction.</p>
Méthode de programmation	<p>Pour programmer l'histoire, l'enfant doit, premièrement, programmer les réactions que le dinosaure aura face à ses illustrations et/ou aux touchés physiques. Ensuite, l'enfant racontera son histoire, en interagissant avec le dinosaure, en plaçant les cartes sur la table interactive ou en touchant les parties du corps du dinosaure.</p>
Tranche d'âge	De 5 à 9 ans
But de l'expérience	<ol style="list-style-type: none"> 1. Observer la prise en main et l'intérêt des enfants pour l'outil. 2. Relever les problèmes techniques de l'outil. 3. Familiariser les enfants avec les notions d'algorithme, de séquence, de programmation d'action et d'utilisation d'actions prédéfinies.
Population touchée	11 enfants (5 filles et 6 garçons). Les enfants travaillent en binôme sauf un qui est seul
Résultats obtenus	<ul style="list-style-type: none"> - L'intérêt que portent les enfants au jeu est très bon. Ils ont tous créé une histoire pendant les 60 minutes de la session de jeu. - La prise en main de l'interface de programmation des actions est facile.
Futur	<p>Améliorer la TUI pour passer au-dessus de quelques limites techniques.</p> <p>Proposer le jeu avec de multiples robots.</p> <p>Mettre en place une nouvelle étude pour voir l'évolution de l'enfant à travers plusieurs sessions.</p>


Quetzal	
Description générale de la TUI	<p>Pour jouer avec cette TUI, l'utilisateur va créer des séquences de programme grâce à des pièces de puzzle en plastique qu'il devra attacher les unes aux autres. Lorsque la séquence d'actions souhaitées est programmée, une caméra capturera celle-ci et la compilera. Le résultat sera affiché par le biais d'un robot LEGO Mindstorms®.</p> 
Input / Output	<p>Input : pièces de puzzle en plastique que l'utilisateur doit attacher les unes aux autres. Elles sont colorées et chaque action est écrite textuellement sur celles-ci. Il existe différents types de pièces de puzzle : des conditions, des boucles, des pièces Start & End séquence, des mouvements. Un système de paramètres et variables est aussi mis au point par le biais de jetons que l'on place dans les tuiles avec une valeur numérique. Nous remarquons aussi la présence d'un spotcode (Code QR) permettant la reconnaissance des pièces de puzzle par la caméra de lecture.</p> <p>Output : le résultat de la séquence formée par les pièces de puzzle est visualisé par le biais du robot qui va exécuter les ordres.</p>
Technologie	<p>Chaque pièce de puzzle est attachée à l'autre de manière physique (I/O) permettant de garder le bon ordre de la séquence. Une caméra digitale raccordée à une tablette ou un ordinateur portable permet de capturer la séquence. Avec cette image, un programme convertit la séquence en un langage intermédiaire. L'ordre de compilation est donné par l'utilisateur via un bouton prévu à cet effet.</p> <p>En cas d'erreur de compilation, celle-ci est indiquée sur l'image grâce à une flèche pointant vers la pièce qui pose problème.</p>
Méthode de programmation	<p>Afin de programmer la séquence à exécuter, l'utilisateur doit assembler des pièces de puzzle les unes aux autres. Chacune d'elles est une action ou une condition. Une caméra lira alors la séquence de pièces de puzzle, la compilera et l'exécutera afin de faire bouger le robot.</p>
Tranche d'âge	7-8 ans
Etudes de l'expérience	<ol style="list-style-type: none">1) Ressortir les problèmes d'utilisation.2) Observer les réactions des utilisateurs face à la programmation.
Population touchée	9 enfants
Résultats obtenus	<ul style="list-style-type: none">- Les enfants ont été capables de lire et construire facilement une séquence.- Tous les enfants n'ont pas été capables de comprendre les effets de leur programme sur le robot.- Certains enfants ont été capables de prédire les erreurs et de les déboguer.
Futur	Etendre les capacités du langage, améliorer les prototypes existants et faire des évaluations plus formelles sur d'autres tranches d'âge.

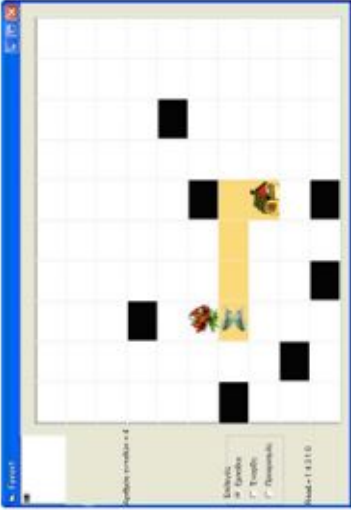

Robo-Blocks	
Description de la TUI	<p>Cette TUI permet à l'utilisateur de prendre le contrôle d'un robot au sol en programmant les séquences de mouvements et d'actions que celui-ci doit réaliser. La programmation des actions se fait grâce à des blocs que l'utilisateur connecte les uns aux autres. En contrôlant le robot, l'utilisateur peut lui faire dessiner des formes ou lui faire suivre un chemin précis. Cette TUI est basée sur le jeu de la tortue géométrique.</p> 
Technologie	<p>La TUI est composée d'un robot qui exécute les commandes programmées, de blocs de commande en acrylique embarqués d'un microcontrôleur, d'un ID unique et d'une connectique RS-232 pour la connexion, d'un bloc principal qui permet le séquençage, l'exécution et la transmission sans fil des instructions au robot.</p>
Input / Output	<p>Input : blocs de commandes que l'on attache les uns aux autres afin de réaliser la séquence. Il existe plusieurs types de commandes : les déplacements (avant, arrière, tourner à droite ou à gauche), un système de variables qui permet de faire tourner ou avancer le robot plus ou moins loin. Pour être exécutée, cette séquence doit être reliée au bloc principal.</p> <p>Output : robot qui montre le résultat, soit en se déplaçant d'un point A vers un point B soit en dessinant la forme demandée.</p>
Méthode de programmation	<p>La programmation de la séquence que le robot doit réaliser se fait grâce aux blocs que l'utilisateur va emboîter les uns aux autres. Afin de programmer les degrés de rotations plus importants, un système d'encodage de variables est prévu.</p>
Tranche d'âge	De 8 à 9 ans.
Etudes effectuées	<ol style="list-style-type: none"> 1) Le test sur les différents modes de jeu sur les enfants de 8-9 ans (le labyrinthe et la réalisation de formes géométriques). 2) L'apprentissage de certains concepts de programmation : débbugging, séquençage, utilisation de variables. 3) La demande d'un feedback des utilisateurs. 4) La mise en place d'une aide à la prise en main et de séances de débbugging selon plusieurs méthodes.
Population touchée	52 enfants de 4 écoles primaires.
Résultats obtenus	<ul style="list-style-type: none"> - Les enfants de moins de 8 ans ont rencontré des difficultés à l'utiliser la TUI. - Les TUI permettent aussi de mettre en avant le débbugging. - Le Débbugging est une étape importante dans la programmation. En effet il permet aux jeunes de comprendre les problèmes rencontrés et de réfléchir sur les solutions possibles à apporter.
Futur	<p>Modifier le débbugging étape par étape par la mise en place de breakpoint (point d'arrêt, méthode qui permet de définir où l'exécution du programme doit s'arrêter)</p> <p>Transformer les blocs de commandes en blocs à choix multiples pour que chaque bloc puisse réaliser chaque action.</p> <p>Permettre un habillage du robot pour le rendre plus attractif.</p>


StarLoop	
Description générale de la TUI	<p>Cette TUI a pour but d'apprendre les sciences informatiques aux enfants. Il s'agit d'un jeu dans lequel les enfants se mettent dans la peau de cadets de l'espace suivant les instructions d'un mentor. Pour réussir le jeu, les enfants sont obligés de collaborer afin de résoudre les problèmes (programmation d'une séquence de mouvements d'un vaisseau spatial). Plusieurs niveaux de difficulté sont disponibles. Pour pouvoir jouer avec cette TUI, les enfants doivent obligatoirement se rendre dans le « space Lab » (pièce numériquement augmentée de 70m²).</p>  
Input / Output	<p>Input : objet tangible que l'utilisateur doit placer sur la table interactive pour transmettre les instructions (de droite à gauche). Il existe 5 types d'actions : mouvement – boucle – procédure – prendre étoile – exécution.</p> <p>Output : le résultat des problèmes résolus sera visible sur les tables interactives ou les murs interactifs du « space Lab ». L'immersion de l'enfant est complète.</p>
Technologie	<p>La TUI prend place dans un local de 70 m² composé de plusieurs appareils d'affichage muraux, de tables interactives, de senseurs pour détecter les manipulations de l'utilisateur, ainsi que de déclencheurs pour répondre physiquement aux interactions de l'utilisateur.</p> <p>Les tables interactives sont de la marque NIKVision et peuvent être utilisées tant de manière indépendante que les unes des autres qu'en collaboration avec le « Space Lab ».</p>
Méthode de programmation	<p>Pour programmer les déplacements du vaisseau, l'utilisateur doit créer des séquences d'actions en alignant les objets tangibles les uns après les autres. Chaque input est une instruction d'actions, de conditions que le programme doit réaliser.</p>
Tranche d'âge	Prévu pour des enfants de 8, 9 ans (3 ^{ème} et 4 ^{ème} primaire)
But de l'expérience	Introduire des concepts de programmation et de la logique informatique aux enfants : séquence – exécution programme - boucles – appel de procédures.
Population touchée	3 Jeunes de 8, 10 et 11 ans, certains ayant déjà une expérience dans la programmation
Résultats obtenus	<ul style="list-style-type: none"> - Pour le plus jeunes de l'équipe, la prise en main n'a pas été facile. Après avoir reçu un peu d'aide, ce dernier a quand même réussi à comprendre le concept de la boucle. - Durant la séance, il n'y a eu aucun problème majeur d'utilisation. La prise en main de la TUI a été jugée relativement facile. - La collaboration entre les enfants est difficile. Le plus âgé devait attendre les plus jeunes. - En conclusion, l'outil est un bon support pour l'apprentissage des concepts de programmation mais le design du jeu est à repenser car la collaboration est difficile.
Futur	Améliorer l'utilisabilité pour améliorer la collaboration entre joueurs.


Strawbies	
Description de la TUI	<p>Cette TUI est un jeu en temps réel dans lequel une framboise doit se déplacer selon une séquence d'actions réalisée par l'utilisateur sous forme de pièces de puzzle. Ce jeu s'inscrit dans un monde ouvert où l'utilisateur devra réaliser des quêtes.</p> 
Technologie	<p>La TUI est composée d'une tablette et de pièces de puzzle en bois. Les séquences d'actions sont formées avec les différentes pièces en bois. Lorsqu'une séquence est construite, la caméra montée sur la tablette lit cette séquence grâce au topcode, la compile et affiche le résultat en faisant bouger le personnage du jeu.</p>
Input	<p>Input : pièces de puzzle en bois que l'utilisateur doit emboîter les unes aux autres. Les actions sont illustrées sur les pièces de puzzle par le biais d'une illustration et d'un topcode (code QR). Il existe différentes actions : déplacement personnage, boucle.</p> <p>Output : tablette sur laquelle le jeu tourne. La caméra montée sur celle-ci va capturer une image de la séquence que l'utilisateur a composée. Via le topcode, le système va compiler et exécuter cette séquence d'actions. Celui-ci affichera le résultat grâce au déplacement du personnage sur l'écran.</p>
Méthode de programmation	Pour programmer une séquence, l'utilisateur doit imbriquer les pièces de puzzle en bois les unes aux autres.
Tranche d'âge	De 5 à 10 ans
Etudes effectuées	<ol style="list-style-type: none"> 1) L'Apprentissage de concepts de programmation : séquence, boucle. 2) L'étude des réactions des enfants face à la TUI : utilisabilité, collaboration
Population touchée	6 sessions de jeu dans deux écoles avec une première fois des enfants de moins de 12 ans et une seconde fois avec des enfants moins de 8 ans.
Résultats obtenus	<ul style="list-style-type: none"> - Les joueurs ayant testé le jeu l'ont apprécié. Ils voulaient continuer à jouer après le temps imparti. - Le jeu permet la collaboration et les négociations sociales.
Futur	/



Tangicons	
Description générale de la TUI	<p>Le jeu est une compétition entre deux groupes de sorciers. Chaque groupe doit construire une tour le plus rapidement possible afin de gagner la partie. Les tangicons permettent aux sorciers de lancer ou casser des enchantements, ce qui provoque un ralentissement de l'évolution de la tour adverse.</p> 
Input / Output	<p>Input : cubes en bois reprenant les idées de base des pièces de puzzle Quetzal. Les actions sur les cubes sont représentées par un topcode et une illustration. Il existe plusieurs types de tangicons : Start and Stop, activer LED, faire un son, ...</p> <p>Output : box LED qui indique les enchantements à rompre via des signaux lumineux et des effets sonores.</p>
Technologie	<p>La TUI est composée des éléments suivants : 12 cubes tangicons, 1 caméra digitale permettant de prendre la photo de la séquence et de la transmettre au système, des briques en bois permettant de créer les tours de jeu, une box composée de LED, le software Quetzal modifié, un ordinateur portable ou fixe pour gérer le système.</p>
Méthode de programmation	<p>Afin de réaliser les enchantements, les utilisateurs de l'équipe 1 doivent créer une séquence d'actions grâce aux tangicons qui va activer les LEDs ou émettre des sons. Et l'équipe 2, pour désenchanter, devra reproduire la séquence de l'enchantement, et ainsi de suite. L'équipe qui est enchantée ne peut pas construire sa tour.</p>
Tranche d'âge	Moins de 6 ans.
Etudes de l'expérience	<ol style="list-style-type: none"> 1) Tester le système mis au point. Il s'agit d'un système qui ne propose pas la résolution d'un problème mais qui prône l'amusement avant tout. 2) Les 5 points principaux que la TUI veut mettre en avant sont : le travail en collaboration, la programmation de séquences, l'utilisation des sens, le raisonnement et la réflexion sur les compétences et les connaissances acquises, l'entraînement des compétences moteurs et senseurs.
Population touchée	2 écoles différentes (16 enfants au total)
Résultats obtenus	<ul style="list-style-type: none"> - Les enfants sont à l'aise avec le jeu et les retours sont très positifs, ils veulent encore y jouer. - Les enfants attendent un peu plus de difficulté. - Les plus gros problèmes proviennent de l'utilisation de la caméra dans le but de photographier la séquence.
Futur	/


TanPro-Kit	
Description générale	<p>Cette TUI est un jeu avec lequel un utilisateur doit faire sortir un personnage d'un labyrinthe en lui faisant effectuer une suite d'actions qu'il détermine en alignant une séquence de cubes. L'utilisateur verra son personnage avancer sur le labyrinthe du LEDpad lors de passage en mode RUN. Le jeu est donc composé de deux grandes phases : la programmation de la séquence et l'exécution de la séquence</p>
Input / Output	 <p>Input : des cubes d'actions qui permettent de définir la séquence que la personne doit réaliser pour sortir du labyrinthe. Il existe 3 types de cubes : cube Start & End, cube de direction et cube senseur. Chaque action est représentée de manière illustrée sur les cubes.</p> <p>Output : le LedPad qui permet de montrer, étape par étape, l'avancement du personnage en fonction de la séquence qu'a généré l'utilisateur avec les cubes d'actions.</p>
Technologie	<p>Les cubes sont composés d'un système d'aimant permettant de les faire tenir les uns aux autres. Ils sont donc alignés de gauche à droite. En plus d'être aimantés, ils sont équipés d'un microprocesseur, d'un module de communication infrarouge et d'un module sans fil. Ces technologies permettent la communication des inputs entre eux et avec le LedPad.</p>
Méthode de programmation	<p>La programmation de la séquence d'actions est faite grâce à l'alignement des cubes. Chaque cube représente une action que le personnage doit effectuer. Lorsque l'utilisateur juge que sa séquence est terminée, il lance le RUN programme.</p>
Tranche d'âge	Enfant de 5 à 9 ans
Etudes effectuées	<ol style="list-style-type: none"> 1) Apprendre les concepts de la programmation : notion de phase de programmation de la séquence et phase d'exécution de la séquence. 2) Juger l'intérêt de l'enfant par rapport à la TUI. 3) Juger la facilité de compréhension et d'utilisation de TanPro-Kit.
Population touchée	16 enfants.
Résultats obtenus	<ul style="list-style-type: none"> - Les enfants apprennent à passer du mode de programmation de la séquence à l'exécution de la séquence pour pouvoir modifier leur programme en fonction des problèmes que le personnage rencontre lors de la séquence. La plupart des enfants ont donc assimilé les deux notions. - Les enfants semblent avoir compris et apprécié le jeu. Ils ont eu une certaine facilité à le prendre en main. - Plusieurs bugs ont été relevés lors de l'expérience.
Futur	<p>Rendre l'outil plus stable et ajouter de nouveaux concepts de programmation à utiliser.</p> <p>Se concentrer sur les feedback d'apprentissage.</p>


T-Butterfly	
Description générale de la TUI	<p>Pour jouer à cette TUI, l'utilisateur doit réaliser des séquences de déplacements en connectant des cubes d'actions les uns aux autres afin de déplacer un papillon virtuel de sa maison (point de départ) à une fleur (point d'arrivée).</p> <div></div>
Input / Output	<p>Input : des cubes de commande à connecter les uns aux autres. Il existe différentes commandes telles que les cubes de mouvements, l'appel d'une procédure de déplacement existante... Chaque action y est représentée sous la forme d'un dessin explicite.</p> <p>Output : le résultat de la séquence formée de cubes par l'utilisateur est visible à l'exécution de celle-ci grâce au déplacement du papillon sur l'écran d'ordinateur.</p>
Technologie	Les cubes mesurent 5*5 cm ou 8.5*2 cm et embarquent un microcontrôleur et des connecteurs bon marché permettant de les joindre au cube principal. Ce dernier est relié à l'ordinateur par un câble RS232.
Méthode de programmation	Pour programmer les mouvements du papillon, l'utilisateur doit connecter les cubes de commandes les uns aux autres et ensuite les relier au cube principal permettant la communication avec le système. Le positionnement des cubes d'action sont identiques au chemin que l'avatar doit parcourir. (Voir image ci-dessus)
Tranche d'âge	Non défini
Etudes de l'expérience	<ul style="list-style-type: none">- Cet article présente un système d'interface tangible utilisateur permettant l'apprentissage de concepts de programmation chez les jeunes : la séquence, l'algorithme.- Aucune étude dans l'article n'est faite. Il s'agit juste d'un article de présentation de la TUI.
Population touchée	Non défini
Résultats obtenus	/
Futur	Enrichir le système avec de nouvelles fonctionnalités et examiner leur impact sur des utilisateurs d'âges et de sexes différents. Examiner les possibilités de pousser la TUI vers un enseignement pour besoins spécialisés : aide au personnel mal voyant ou ayant des problèmes psychomoteurs.

Tern en milieu scolaire	
Description générale de la TUI	<p>Pour jouer avec cette TUI, l'utilisateur va créer des séquences de programme grâce à des pièces de puzzle en bois qu'il devra attacher les unes aux autres. Lorsque la séquence d'actions est programmée, une caméra capturera celle-ci et la compilera. L'utilisateur verra le résultat par le biais d'un écran d'ordinateur.</p> 
Input / Output	<p>Input : pièces de puzzle en bois que l'utilisateur doit attacher les unes aux autres. Elles sont colorées et chaque action est écrite textuellement sur celles-ci. Il existe différents types de pièces de puzzle : des conditions, des boucles, des puzzles de Start & End séquence, des mouvements, des appels de procédures. Un système de paramètres est aussi disponible. Nous remarquons également la présence d'un spotcode (Code QR) permettant la reconnaissance des pièces de puzzle par la caméra de lecture.</p> <p>Output : le résultat de la séquence formée par les pièces de puzzle est visualisé sur l'écran d'ordinateur.</p>
Technologie	Chaque pièce de puzzle est attachée l'une à l'autre de manière physique (I/O) permettant de garder le bon ordre de la séquence. Une caméra digitale raccordée à une tablette ou à un ordinateur portable permet de capturer la séquence. Avec cette image, un programme converti la séquence en un langage de programmation. L'ordre de compilation est donné par l'utilisateur via un bouton prévu à cet effet.
Méthode de programmation	Afin de programmer la séquence à exécuter, l'utilisateur devra assembler les pièces de bois les unes aux autres afin de réaliser une séquence d'actions. Chaque pièce de puzzle assemblée est une action, une condition que le programme doit respecter. Une caméra lira alors la séquence, la compilera et l'exécutera afin de faire bouger le robot sur l'écran d'ordinateur.
Tranche d'âge	7-8 ans
Etudes de l'expérience	<ol style="list-style-type: none"> 1) Ressortir les problèmes d'utilisation. 2) Observer les réactions des utilisateurs face à la programmation physique.
Population touchée	9 enfants
Résultats obtenus	<ul style="list-style-type: none"> - Les enfants ont été capables de lire et de construire facilement une séquence. - Tous les enfants n'ont pas été capables de comprendre les effets de leur programme sur le robot. - Certains enfants ont été capables de prédire les erreurs et de les déboguer.
Futur	Etendre les capacités du langage, améliorer les prototypes existants et faire des évaluations plus formelles sur d'autres tranches d'âge.

Tern dans un musée	
Description générale de la TUI	<p>Cette expérience se déroule dans un musée. Toute personne voulant l'essayer doit juste s'arrêter et attendre son tour. Il s'agit d'un système composé de pièces de puzzle en bois que l'on doit placer les unes après les autres afin de créer une séquence qu'un robot reproduira pour aller d'un point x vers un point y.</p> 
Input / Output	<p>Input : pièces de puzzle en bois qui s'attachent les unes aux autres. Sur chaque pièce est indiquée textuellement l'action et le TopCode (Code QR). Il existe 7 types de pièces différentes : motion, dance, sound, control, sensor, nombre paramètre, logic. Chaque type de pièce est d'une couleur différente.</p> <p>Output : L'output est le robot qui reproduira la séquence faite par l'utilisateur.</p>
Technologie	<p>Cette TUI est composée de pièces de puzzle en bois et d'une caméra digitale permettant la capture de la séquence physique. Grâce à la lecture des topcodes présents sur l'image capturée par la caméra, le programme va pouvoir compiler et exécuter la séquence générée par l'utilisateur en la transmettant au robot pour qu'il s'active. Le résultat de la séquence sera donc visible par le biais du robot qui se déplace.</p>
Méthode de programmation	<p>Pour programmer la séquence à exécuter, l'utilisateur doit assembler les pièces de puzzle les unes aux autres. Chaque pièce de puzzle est une action, une condition que le programme doit respecter. A l'exécution, la caméra capture la séquence, la compile et l'exécute.</p>
Tranche d'âge	<p>Non défini car la TUI est exposée dans un musée. Objectif : essayer d'attirer des enfants d'écoles primaires avec leurs parents.</p>
Etudes de l'expérience	<ol style="list-style-type: none"> 1) L'attractivité d'un tel appareil dans un musée. (incitation à jouer à la TUI) 2) Facilité de la prise en main de la TUI. 3) Le temps de concentration. 4) Support d'une interaction en groupe. 5) La maintenance de la TUI.
Population touchée	<p>Tout public : essayer d'avoir autant de filles que de garçons et de tout âge. (environ 700 à 800 personnes sur le mois)</p>
Résultats obtenus	<ul style="list-style-type: none"> - La taille moyenne des groupes utilisant la TUI est de 2.4 personnes. - La prise en main est complexe puisqu'elle est de 52.6% lors de la 1^{ère} session de test, et de 68.6% au cours de la 2^{ème} session. - Les groupes sont restés en moyenne moins de 5 minutes 20 secondes par session. - Au cours des sessions tests, il y a eu quelques problèmes avec la caméra et le robot.
Futur	<p>Réaliser plus d'expériences pour comprendre le processus d'apprentissage des visiteurs</p>

T-Maze	
Description générale de la TUI	<p>Cette TUI est un jeu de labyrinthe. Le but est de faire sortir une personne d'un labyrinthe grâce à des séquences programmées par des cubes et l'utilisation de senseurs. Après la réalisation de la séquence à appliquer, l'utilisateur verra son personnage évoluer sur l'écran d'ordinateur. Il lui est aussi donné la possibilité de créer ses propres labyrinthes.</p>
Input / Output	<div></div> <p>Input : cubes d'actions aimantés (30mm*30mm*30mm), à placer en dessous d'une caméra (sens de lecture de gauche à droite). Ces cubes permettent de définir la séquence que le personnage doit réaliser pour sortir du labyrinthe. Il en existe 4 types différents : action start & end, cube de direction, boucle et senseurs. Chaque action est représentée de manière illustrée sur les cubes.</p> <p>Output : l'écran d'ordinateur permet de montrer, étape par étape, l'avancement du personnage en fonction de la séquence qu'a généré l'utilisateur avec les cubes d'actions.</p>
Technologie	<p>Les cubes d'action sont en bois, une illustration de type TopCode (Code QR) y est dessinée. Grâce à un système de caméra, une image de la séquence est capturée. Le programme lira et compilera cette séquence en définissant les actions à entreprendre grâce au code QR dessiné sur les cubes. Ensuite, la TUI affichera le résultat sur l'écran d'ordinateur.</p>
Méthode de programmation	<p>Comme pour les styles précédents, l'utilisateur programme la séquence d'actions en alignant des cubes les uns après les autres. Chaque cube placé est une action que doit réaliser l'avatar. Lorsque la séquence comporte une action senseur, l'avatar s'arrête dans ses déplacements tant que l'utilisateur n'aura pas pressé le senseur correspondant. La caméra capture l'image et exécute la séquence programmée par l'utilisateur.</p>
Tranche d'âge	De 5 ans à 9 ans
Etudes de l'expérience	<ol style="list-style-type: none">1) Le niveau d'intérêt de l'enfant pour le jeu.2) La facilité de la prise en main de la TUI avec le placement des cubes et la lecture de ceux-ci via la caméra.3) L'apprentissage de concepts de programmation : séquence, variable, boucle, Debugging (real time feedback).4) L'augmentation de l'intérêt de l'enfant aux sciences modernes et technologiques via le jeu des senseurs.5) La conscientisation de l'enfant à optimiser son résultat.
Population touchée	10 enfants (3 garçons et 7 filles, 90% ont déjà utilisé un ordinateur)
Résultats obtenus	<ul style="list-style-type: none">- jouabilité : les enfants ont semblé très intéressés mais ils ont rencontré des difficultés avec les boucles ;- utilisabilité et apprentissage : pour les plus jeunes, la prise en main est un peu plus difficile à cause de l'utilisation de la caméra. Mais lorsque des explications sont données, l'enfant est à l'aise face à la TUI ;- apprentissage des notions de programmation : Les enfants de cet âge peuvent être intéressés mais le concept de boucle n'est pas facile même si certains tentaient de l'utiliser pour optimiser la sortie du labyrinthe.
Futur	Améliorer le design, augmenter la panoplie de senseurs, ajouter des blocs différents (condition logique, etc.), développer plus de jeu et scénarii.

Toque	
Description de la TUI	<p>Il s'agit d'une TUI où l'utilisateur va pouvoir se mettre à la place d'un cuisinier. L'objectif du jeu consiste à réaliser des recettes de cuisine en indiquant les actions à mener : sélectionner un ingrédient, couper celui-ci... Pour ce faire, les utilisateurs jouent avec un système de manettes de Wii : la Wiimote, et le Nunchuk afin de contrôler le cuisinier.</p> 
Technologie	<p>Cette TUI est composée d'une partie avec écran d'ordinateur en guise d'output, une Wiimote et un Nunchuk en guise d'input. Ils sont connectés via Bluetooth pour rentrer dans la peau d'un cuisinier et ainsi réaliser des recettes.</p>
Input / Output	<p>Input : la Wiimote et le Nunchuk qui permettent de générer les séquences d'actions. Les utilisateurs ont aussi la possibilité de réaliser les séquences grâce à des cartes représentant les actions.</p> <p>Output : l'écran d'ordinateur sur lequel chaque action commandée par l'utilisateur sera retransmise.</p>
Méthode de programmation	<p>L'utilisateur a deux possibilités :</p> <ul style="list-style-type: none"> - programmer les étapes de la recette à l'aide des manettes en réalisant les actions (gestuelles ou de sélection) les unes à la suite des autres dans un ordre bien précis dicté par l'application qui oriente l'utilisateur ; - programmer les étapes de la recette grâce à des cartes représentant les actions à réaliser. Les utilisateurs peuvent aussi dessiner les actions à réaliser eux-mêmes.
Tranche d'âge	De 7 à 11 ans.
Etudes effectuées	<ol style="list-style-type: none"> 1) Travail avec des jeunes de 7 à 11 ans dans le but d'explorer le prototype et d'améliorer le design de la TUI. 2) Apprentissage de certains concepts de programmation : objet, méthodes, arguments, boucle... 3) Travail en collaboration
Population touchée	9 enfants (5 filles et 4 garçons)
Résultats obtenus	<ul style="list-style-type: none"> - Les enfants travaillaient ensemble grâce à la Wiimote et au Nunchuk. - Le concept de boucle est complexe à comprendre pour les enfants. Ils trouvent les boucles lassantes et préféreraient la programmation en parallèle.
Futur	<p>Améliorer Toque pour qu'il prenne en charge la programmation en parallèle.</p> <p>Rendre la TUI plus flexible dans la réalisation des séquences d'actions.</p> <p>Comparer la programmation tangible seule avec la programmation tangible en collaboration</p>

T-ProRob	
Description générale du TUI	<p>Pour utiliser cette TUI, l'utilisateur doit connecter des blocs de commandes en plexiglas les uns aux autres afin de créer une séquence. Celle-ci permet de définir ce que le robot NXT Lego doit réaliser.</p> 
Input / Output	<p>Input : blocs en plexiglas que l'on connecte les uns aux autres de gauche à droite. Il existe différents types de blocs tels que les cubes de mouvements, les cubes de boucles, les cubes de conditions, les cubes de senseurs, les cubes d'appel d'une fonction. Chaque action y est représentée sous forme d'un dessin explicite ou sous forme textuelle pour les paramètres/variables. Le poids des blocs varient en fonction de leur action ou valeur.</p> <p>Output : le résultat de la séquence des blocs formée par l'utilisateur est visible à l'exécution du robot. Un feedback d'exécution est donné via un système de LED sur les cubes.</p>
Technologie	<p>Les blocs sont des microcontrôleurs reliés entre eux avec des connecteurs bon marché (D9 & D25). Une différence entre les connecteurs pour les paramètres/variables et les connecteurs des blocs d'action a été faite pour permettre de réduire le nombre de fautes de syntaxe. Un bloc spécial permet la communication entre les inputs et le système via Bluetooth.</p>
Méthode de programmation	<p>Pour programmer les mouvements du robot, l'utilisateur doit réaliser des séquences d'actions en alignant les blocs les uns à la suite des autres. Chaque cube représente une action.</p>
Tranche d'âge	<p>Enfants de 6 à 12 ans répartis par groupe d'âge.</p>
Etudes de l'expérience	<ol style="list-style-type: none"> 1) L'observation des réactions des utilisateurs face à une TUI ou GUI (identique sauf leur mode de fonctionnement) et de conclusions sur les points forts et faibles de chacun. 2) L'apprentissage de certains concepts de programmation : la planification d'une séquence, son exécution et le débogging, la mise en place de conditions et de boucles, l'utilisation de paramètres et l'appel de procédure(s).
Population touchée	<p>20 enfants de 6-7 ans / 25 enfants de 7-8 ans / 14 enfants de 9-10 ans / 25 enfants de 10-11 ans / 25 enfants de 11-12 ans</p> <p>Total : 109 enfants.</p>
Résultats obtenus	<ul style="list-style-type: none"> - Sur les deux interfaces, le temps d'accomplissement de la tâche croît avec l'âge qui augmente. - Les enfants font plus d'erreurs de programmation sur le GUI (Graphical User Interface) que sur la TUI car ils sont plus actifs et motivés sur la TUI. - L'utilisation d'une TUI facilite le débogging par rapport au GUI. - Sur la TUI, les enfants prennent plus de temps durant leur temps libre pour essayer et pour comprendre. - Les enfants plus jeunes n'utilisent pas assez les blocs d'actions de type « boucles – conditions ». Ils restent sur une syntaxe plus simple.
Futur	<p>Examiner les possibilités de pousser la TUI vers un enseignement pour besoins spécialisés : aide au personnel mal voyant ou ayant des problèmes psychomoteurs.</p> <p>Essayer d'ajouter des techniques de programmation moderne (Pair programming, notion d'objet ...)</p>

Turtan	
Description générale de la TUI	<p>Il s'agit d'une table avec une surface interactive sur laquelle on place des instructions qui font réagir une tortue. Tout se fait en simultané, pas de distinction entre le codage et l'exécution de l'application. L'utilisateur place les actions sur la table interactive et ensuite le résultat s'affiche selon les instructions commandées. Dans le cas de cette TUI, une tortue se déplace en dessinant des traits sur la surface interactive.</p>
Input / Output	<p>Input : objet tangible de type bloc/cube sur lequel est représenté un dessin d'une action à réaliser. Il existe différents types d'actions sur les cubes : les déplacements avec ou sans peinture, les rotations gauche/droite, les boucles, les changements de grandeur de trait, le <u>start</u>, les changements de couleurs.</p> <p>Output : la table interactive permettant de visualiser le résultat des instructions placées par l'utilisateur.</p>
Technologie	Table interactive tangible utilisant un projecteur et une caméra sous la table. La caméra capture chaque instruction placée par l'utilisateur sur la table. Le système l'identifie et ensuite, via le projecteur, transmet le résultat (déplacement de la tortue) sur la table. Le système est donc composé d'un PC avec 2 processeurs séparés : le 1 ^{er} gère la caméra et le 2 ^{ème} s'occupe de calculer et d'afficher le résultat.
Méthode de programmation	Pour programmer les mouvements de la tortue, l'utilisateur doit poser sur la table interactive les inputs. Chaque input posé correspond à une instruction que le programme exécutera. L'utilisateur a aussi la possibilité de modifier les trajectoires, les longueurs, la grosseur des lignes en tournant l'input d'action correspondant sur la table interactive.
Tranche d'âge	Non défini
Etude de l'expérience	Présenter et explorer les concepts de programmation : « Séquence / boucle »
Population touchée	Enfant n'ayant aucune notion de programmation.
Résultats obtenus	<ul style="list-style-type: none"> - Pas de tests probants pour vérifier la valeur d'apprentissage - Des tests ont été réalisés pour améliorer des problèmes de design et d'implémentation. - Compréhension facile pour l'utilisateur / l'interaction avec la TUI a été appréciée.
Futur	Tester sur un échantillon d'enfants pour mesurer l'impact du niveau d'apprentissage.

5.1 Classifications

Après avoir posé la problématique, d'avoir pris le temps de comprendre et de s'imprégner du sujet, cette partie du mémoire va présenter les différentes classifications qui ont été mises sur pied dans le but d'essayer de répondre à la question de recherche.

Trois classifications vont être présentées, tout d'abord, la classification de Yu va être présentée, ensuite celle selon la taxonomie de Fishkin et pour finir, une dernière classification réalisée dans le cadre de ce mémoire.

5.1.1 Classification selon Yu [40]

La classification de Yu propose une catégorisation des TUI sur base de leurs fonctionnalités physiques. Les catégories sont les suivantes :

- physique avec électronique : un dispositif est dit physique avec électronique quand celui-ci n'est composé que d'élément tangible avec électronique embarquée
- physique sans électronique : un dispositif est dit physique sans électronique quand celui-ci n'est composé uniquement que d'élément tangible sans électronique embarquée
- virtuel : un dispositif virtuel est un dispositif qui n'existe pas physiquement
- hybride avec des blocs de programmation virtuel : Un dispositif est appelé hybride lorsqu'il comporte à la fois des éléments virtuels et physiques mais les blocs de programmation sont virtuels
- hybride avec des blocs de programmation tangible : Un dispositif est appelé hybride lorsqu'il comporte à la fois des éléments virtuels et physiques mais les blocs de programmation sont tangible

	Physique sans électronique	Physique avec de l'électronique	Virtuel	Hybrides avec des blocs de programmation virtuel	Hybrides avec des blocs de programmation tangibles
Nom des dispositifs		Bee-bot Dr. Wagon TantPro-Kit Electronic Blocks KIBO Robo-Blocks			Algoblock Dialando Pleo Quetzal StarLoop Strawbies Tangicons E-Blocks T-Butterfly Tern T-Maze Toque T-ProRob Turtant

FIGURE 1 – Classification de Yu

Si on classe les vingt outils sélectionnés, on constate qu'ils sont regroupés dans deux des cinq catégories : physique avec électronique, pour six sur les vingt, et hybrides avec des blocs de programmation tangible pour les quatorze restants.

5.1.2 Classification selon Fishkin

Pour rappel, cette classification permet de trier les différents dispositifs selon deux grands axes : leur incarnation qui juge le degré de distinction qu'il y a entre l'input et l'output, et la métaphore, qui juge la relation entre l'action de l'utilisateur sur la TUI et cette-même action dans le monde réel.

Incarnation	Métaphore			
	Nom	Verbe	Nom & verbe	Complète
Complète		Electronic Blocks, Pleo	Bee-bot	
Proche		Electronic Blocks, KIBO, StarLoop, Turntan		
Environnement				
Distante		Algoblock, Dialando, Dr. Wagon, E-Block, KIBO, Pleo, Quetzal, Robo-Blocks, Strawbies, Tangicons, Tan-Pro Kit, T-Butterfly, Tern, T-Maze, T-ProRob, Toque		

FIGURE 2 – Tableau de Fishkin

Si l'on classe les différents outils selon les critères de cette taxonomie, on constate qu'en terme de métaphore d'inputs, les outils sont catégorisés en deux groupe : verbe pour la majorité des outils (dix-neuf sur les vingt) et nom & verbe pour le dernier. En ce qui concerne les incarnations, les outils sont catégorisé en trois catégories : complète (trois sur vingt), proche (quatre sur vingt) et distante (seize sur vingt)

5.1.3 Classification selon un tableau de critères originaux

Dans un premier temps, j'ai mis sur papier des critères qui me paraissaient intéressants à comparer : le sens de lecture des INPUTS, le format des INPUTS, la relation entre les couleurs d'inputs et les actions qu'ils provoquent, le système de communication entre INPUT/SYSTEME, le mode d'affichage du résultat, le nombre d'enfants ayant réalisé l'expérience, ...

Le nombre de critères trop important a rendu le tableau illisible. Il était alors impossible d'en tirer des conclusions compréhensibles. Une généralisation a été faite par la suite en gardant les critères jugés les plus pertinents, permettant au tableau de devenir lisible.

Définitions des critères

Présentation des différents critères composant le tableau :

1. CONTEXTE DE L'EXPERIENCE

0 Non décrit dans l'article analysé

— Critères A : Age

1 De 2 ans à 6 ans.

2 De 7 ans à 12 ans.

L'âge maximum des enfants est limité à 12 ans. La majorité des outils ont été testés en milieu scolaire primaire. Ces deux catégories proviennent d'une division selon la théorie de « Piaget » et les compétences d'apprentissage de l'enfant selon son âge.

— Critère B : Cadre

1 Interne à l'école : lors de l'expérimentation, la TUI a été utilisée dans un milieu scolaire.

2 Externe à l'école :

a Musée : lors de l'expérimentation, la TUI a été utilisée dans un musée.

b Laboratoire : la TUI a été expérimentée dans un laboratoire.

Ce critère du cadre permet de connaître le lieu dans lequel les tests de la TUI ont été effectués.

2. ASPECTS PEDAGOGIQUES

— Critère C : Modalités de travail

1 Individuel : l'utilisation du TUI au cours de l'étude se fait de manière individuelle.

2 En groupe : travail en collaboration. L'utilisation de la TUI au cours de son expérimentation a été réalisée en groupe.

Ce critère permet de mettre en avant la mise en place d'un travail collaboratif, permettant de renforcer les liens sociaux entre les utilisateurs.

— **Critère D – Complexité :**

- 1 Un seul niveau : la TUI ne propose qu'un niveau de difficulté.
- 2 Plusieurs niveaux : la TUI propose plusieurs niveaux de difficulté.

Ce critère permet d'avoir une idée d'une possible utilisation de la TUI sur une tranche d'âge plus élevée.

3. ASPECTS DIDACTIQUES

— **Critère E : Concepts de programmation mis en œuvre par l'utilisateur**

- 1 Algorithme / Programme (sans boucle infinie)
- 2 Structure de contrôle séquentielle (séquence)
- 3 Structure conditionnelle
- 4 Boucle
- 5 Variable
- 6 Sous-programme

Ce critère va permettre de connaître les concepts de programmation que l'utilisateur devra utiliser pour arriver à la résolution de l'exercice.

4. INTERACTION

— **Critère F : Dispositif physique**

- 1 Tout-en-un : l'input et l'output sont un seul élément.
- 2 Composé : Les inputs et les outputs de la TUI sont séparés en plusieurs éléments.
Exemple : les inputs sont des blocs de programmation et l'output est un écran d'ordinateur, comme pour le T-Maze.
- 3 Input : ce critère va permettre de définir la méthode d'interaction qu'il y a entre l'utilisateur et la TUI au niveau des INPUTS. Cette méthode de classification est basée sur celle de Roudaut [19]
 - a Continu : il s'agit d'une interaction par mouvement. Un dispositif est dit « continu » s'il récupère un signal continu représentant la position d'un pointeur (Exemple : pointeur de souris sur l'écran d'ordinateur, position du doigt sur l'écran tactile).
 - b Discret : il s'agit d'une interaction par mouvement. Un dispositif discret récupère une valeur booléenne (exemple : un bouton actionné ou non).
 - c Embarquée : il s'agit d'une interaction par mouvement. Une interface embarquée utilise le dispositif lui-même comme Input grâce à divers capteurs (détecteur de mouvement, inertielle, capteur de déplacements ou dispositif d'identification unique...).

- d** Vocale : il s'agit d'une interaction qui se fait grâce à des microphones.
 - e** Autres capteurs (environnement) : il s'agit d'une interaction passive.
Le dispositif, grâce à ses capteurs internes et externes, s'adapte à son environnement sans action spécifique de l'utilisateur.
- 4** Output : ce critère va permettre de définir la méthode d'interaction qu'il y a entre l'utilisateur et la TUI au niveau des OUTPUTS. Cette méthode de classification est basée sur celle de Roudaut [19]
 - a** Ecran : interaction visuelle.
 - b** LED : interaction visuelle.
 - c** Hauts-parleurs : Interaction auditive.
 - d** Dispositifs haptiques : interaction kinesthésiques
 - e** Déplacement robot : interaction visuelle.
- **Critère G : Action de l'utilisateur sur les inputs**
 - 1** Paramétrage (prédéfini) : les inputs que l'utilisateur peut utiliser sont prédéfinis.
 - 2** Personnalisation : l'utilisateur a le droit de personnaliser ses inputs pour, par exemple, permettre d'ajouter des actions.

Présentation de la classification

CONTEXTE de l'expérience										ASPECTS PÉDAGOGIQUES					ASPECTS DIDACTIQUES					INTERACTIONS																	
A					B					C		D			E					F										G							
1	2	0	1		2				2		1	2	0	1	2	0	1	2	3	4	5	6	1	2	3					4					1		2
				a	b				1																a	b	c	d	e	a	b	c	d	e			
1	AlgoBlock	x			x				x				x	x	x	x	x						x						x					x			
2	Bee-Bot	x		x					x				x	x	x	x						x													x		
3	Dialando																																			x	
4	Dr. Wagon	x	x						x				x	x	x	x						x													x		
5	E-Block	x	x						x				x	x	x	x																			x		
6	Electronic Blocks	x																																		x	
7	KIBO	x																																		x	
8	Pleo	x	x						x																											x	
9	Quetzal																																			x	
10	Robo-Blocks																																			x	
11	StarLoop	x	x						x																											x	
12	Strawbies	x	x						x																											x	
13	Tangicons	x																																		x	
14	Tan-Pro Kit	x	x																																	x	
15	T-Butterfly																																			x	
16	Tern																																			x	
17	T-Maze																																			x	
18	Toque																																			x	
19	T-ProRob	x	x																																	x	
20	TurTan																																			x	

0. Non décrit

A. Âge

1. De 2 à 6 ans

2. De 7 à 12 ans (inclus)

B. Cadre

1. Interne à l'école

2. Externe à l'école

a. Musée

b. Laboratoire

C. Modalités de travail

1. Individuel

2. En groupe

a. Collaboration

D. Complexité

1. Un seul niveau

2. Plusieurs niveaux

E. Concepts de programmation mis en œuvre

1. Algorithme/Programme (fini)

2.Séquence

3. Structure conditionnelle

4. Boucle

5. Variable

6. Sous-programme

F. Dispositif physique

1. Tout-en-un

2. Composé

3. Input

a. Continu

b. Discret

c. Interaction embarquée (mouvement)

d. Vocale

e. Autres capteurs (environnement)

4. Output

a. Écran

b. Led

c. Haut-parleurs

d. Dispositifs haptiques

e.Déplacement robot

G. Action de l'utilisateur sur les inputs

1. Paramétrage (prédéfini)

2. Personnalisation

FIGURE 3 – Tableau de critères originaux

Si l'on classe les différents outils, on peut constater que d'un point de vue contexte de l'expérience menée par les chercheurs, l'âge requis pour l'utilisation des TUI est plutôt bien réparti : douze dispositifs pour les moins de sept ans et douze pour les plus de sept ans. Le cadre dans laquelle l'expérience a été menée est, quant à lui, pour sept outils sur les vingt en milieu scolaire, cinq des outils ont été testés en laboratoire et un a trouvé place dans un musée.

Au niveau des aspects pédagogiques, le travail en collaboration est mis en avant pour huit outils, les douze derniers sont, quant à eux, indéfinis. L'autre critère pédagogiques est la notion de niveau de difficulté pour l'utilisation du dispositif, cinq des vingt dispositifs proposent plusieurs niveaux de difficulté contre un seul n'en proposant qu'un, pour les autres dispositifs, l'information n'est pas mentionnée.

L'aspect didactique reprend les concepts de programmation abordés par les différents dispositifs : tous les outils abordent la notion d'algorithme, dix-neuf la notion de séquence, six dispositifs donnent la possibilité d'utiliser les structures conditionnelles, douze abordent la boucle, le concept de variable est abordé par 6 dispositifs et pour terminer, la notion de sous-programme est abordé par quatre dispositifs seulement.

Pour terminer, grâce à l'aspect interaction, on peut constater que la majorité des dispositifs, dix-huit sur vingt, sont « composé ». Seulement deux dispositifs sont « tout-en-un ». Au niveau des interactions input et utilisateur, l'interaction qui est la plus représentée est la « passive » (dix-huit fois), la « continue » est proposée par deux dispositifs, sept proposent une interaction « discrète », trois sont de type « embarqué » et une seule « vocale ». Les interactions output utilisateur sont majoritairement de type « visuel » : onze via un écran, dix utilisent les LED, quatre le déplacement d'un robot. Pour finir on voit que l'interaction « sonore » est utilisée huit fois et que « l'haptique » une seule fois.

Chapitre 6

Analyse et discussion

L'analyse et la discussion qui suit fait l'objet d'une publication pour la 30e Conférence Francophone sur l'Interaction Homme-Machine. [9]

L'analyse de la classification de Yu (Figure 6.1) révèle que les vingt outils pour l'apprentissage des concepts de programmation chez les enfants, sont répartis en deux grandes catégories : « Physique avec électronique » (six TUI sur vingt) et « Hybride avec des blocs de programmation tangible » (quatorze TUI sur vingt). La seconde est donc la plus représentée.

Cette constatation est logique puisque la majorité des jeux, comme T-Maze [35], sont des jeux dans lesquels l'utilisateur doit mouvoir un avatar d'un point A vers un point B dans un univers imaginaire. La programmation des mouvements se fait via des blocs d'actions tangibles embarqués ou non en bois, en plastique ou en métal. Le résultat de ces mouvements est affiché sur un élément à part entière de manière virtuelle (écran, tablette, etc...) d'où l'appellation « hybride ». Cependant, il existe, une exception à ce genre de jeu : TanPro-Kit [34], qui, lui, entre dans la catégorie des dispositifs physiques avec électronique. L'aspect virtuel du résultat est remplacé par l'utilisation d'un LedPad (plateau de jeu composé de Leds qui permettent d'afficher l'emplacement de l'avatar sur le labyrinthe). La programmation des mouvements se fait via des blocs d'actions tangibles embarqués.

Les cinq autres outils formant la catégorie des dispositifs physiques avec de l'électronique sont, quant à eux, principalement des outils incluant un robot. Trois d'entre eux présentent aussi une programmation de déplacement via des blocs.

Les blocs de programmation tiennent donc une place importante puisque qu'ils sont présents dans dix-sept TUI sur vingt. Cette classification générale pourrait sans doute être améliorée si on y ajoute un aspect de description des blocs de programmation.



FIGURE 1 – Tern / Strawbies - Contrainte syntaxique

Par exemple, on pourrait y ajouter des contraintes syntaxiques (figure 7.1) comme le proposent les outils Tern [12] ou Strawbies [14] avec la forme de leurs inputs, ou par l'ajout simplement de l'aspect embarqué ou non du bloc de programmation. Une autre possibilité serait de représenter les concepts de programmation de manière physique par les blocs comme dans l'outil Dr. Wagon [2]. Ce dernier essaye de mettre en place cette représentation en englobant les actions présentes dans le concept de boucle.

Le tableau reprenant les outils classés selon la taxonomie de Fishkin (figure 6.2) permet de mettre en avant le travail en métaphore verbale de la majorité des TUI (dix-neuf sur vingt). Il s'agit principalement de TUI, comme Tangicons[23], T-Maze[35], E-Block[36] utilisant comme input des blocs de programmation représentant des actions telles que : avancer, tourner à gauche ou droite, allumer une LED, émettre un son, etc. . .

Un outil se démarque de cette classification à sens unique. Il s'agit de l'outil Bee-bot [15] qui présente une métaphore Verbe & nom grâce à son système de flèches directionnelles directement placées sur le dos de l'appareil que l'enfant doit appuyer pour programmer les mouvements de l'abeille.

Du point de vue des différentes incarnations, les outils se catégorisent en 3 groupes : les incarnations dites « complètes » comme le représente très bien Bee-Bot [15], les incarnations dites « proches » à l'instar de Turtan[6] et StarLoop [18] qui travaillent sur des tables interactives, et pour terminer les incarnations dites « distantes » qui affichent les résultats sur des dispositifs à part comme des écrans ou tablette à côté des inputs. Les jeux comme T-Maze [35], Tanpro-Kit [34] en sont les parfaits exemples. (figure 7.2)

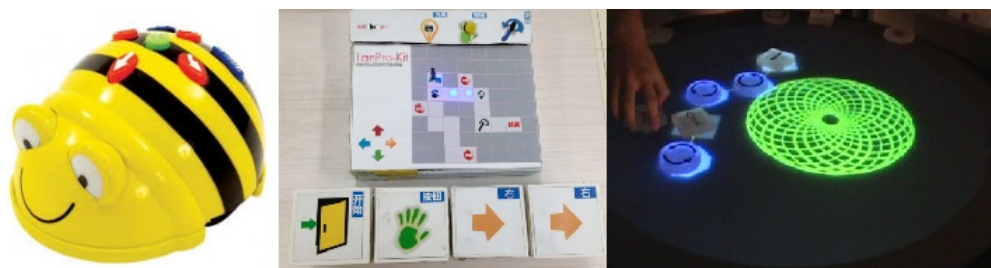


FIGURE 2 – Bee-Bot / TanPro-Kit / Turtan - Incarnation

Il est important de constater que la classification de Fishkin se veut exclusive. Ce qui veut dire que chaque outil doit normalement se retrouver dans une seule catégorie.

Cependant, plusieurs outils ont été classés dans deux catégories : Electronic Blocks [37, 38], KIBO [27] et Pleo [20].

Electronics Blocks propose une incarnation complète et proche : complète si l'on considère les blocs assemblés comme un unique dispositif et proche si chaque bloc, pris individuellement et mis en connexion l'un avec l'autre, provoque une réaction. En revanche, Kibo est d'interaction proche et distante car son appareil interne doit scanner les inputs d'action et ensuite, les actions sont exécutées par le robot de manière distante. Pour finir, Pleo est à la fois d'incarnation complète et distante car elle possède de la programmation physique sur l'outil ainsi que virtuelle sur la tablette.

Une première constatation, touchant les critères d'âge et le cadre de l'expérience, faite grâce au tableau de critères originaux (Figure 6.3), est qu'il existe un lien entre l'âge d'utilisation de la TUI et son aspect physique. La majorité des TUI touchant des utilisateurs de moins de sept ans possède le critère « physique avec électronique » selon la classification de Yu et « tout-en-un », comme le montre bee-bot, KIBO et les electronic blocks. Ceci est sûrement dû à la facilité avec laquelle les enfants se concentrent et prennent en main une TUI d'un seul élément (des cubes à emboîter ou un robot) plutôt que de devoir jongler avec plusieurs éléments (un écran, des blocs tangibles et/ou une caméra, etc.).

Une deuxième constatation peut être faite avec l'âge d'utilisation et les concepts de programmation que la TUI permet d'utiliser. Plus les enfants sont jeunes, moins ils utiliseront des concepts de programmation complexe tel que la boucle. Ils resteront sur des séquences de programmation d'actions simples comme l'indique les résultats de l'expérience T-Maze[35].

Une troisième constatation est ressortie lors la lecture du tableau. Certaines TUI mettent en place plusieurs niveaux de difficulté dans le but de toucher une population plus large en adaptant les concepts de programmation à utiliser. En effet, plus l'enfant est âgé, plus les concepts de programmation abordés seront complexes à gérer. Par exemple, les premiers niveaux utilisent des concepts de programmation simple comme une séquence programmant des déplacements : gauche, droite, avant, arrière. Au fur et à mesure que l'enfant atteint les objectifs, la TUI proposera de passer au niveau supérieur en incorporant de nouvelles notions de programmation plus complexes comme la structure conditionnelle, la boucle et l'utilisation de variables.

Au point de vue pédagogique, l'utilisation des TUI dans les écoles permet de mettre en évidence le travail en collaboration. Des chercheurs pensent se servir, dans le futur, des TUI pour apprendre les bases du travail en « Pair Programming » (programmation en duo) [22].

Un autre aspect, qui n'est pas repris dans le tableau mais qui est souvent mis en avant, est la concentration de l'enfant. Il ressort que les utilisateurs restent concentrés plus longtemps en utilisant une TUI plutôt qu'un GUI¹ car ils manipulent des objets tangibles. L'article [22] montre des tableaux de chiffres appuyant ce résultat.

1. Graphical User Interface

De plus, la manière d'utiliser la TUI durant le temps libre des enfants est mis en avant. Effectivement, ils laissent libre cours à leur imagination en essayant des séquences de programmation qu'ils ne réaliseraient pas durant les sessions de jeu surveillées. Cette constatation permet l'autodidaxie des concepts et leur assimilation.

En ce qui concerne la partie didactiques du tableau (Figure 6.3), on remarque que les concepts de programmation qui sont abordés dans les TUI sont au nombre de six : l'algorithme, la séquence, l'utilisation de structures conditionnelles, la boucle, la variable et le sous-programme. Une analogie avec les notions de programmation abordées dans le cours « d'introduction à la programmation » peut être faite [32]. Celle-ci démontre que le cours, dispensé à l'Université de Namur, propose d'aborder davantage de concepts que ne le permettent les TUI.

La notion d'algorithme est présente dans l'utilisation de toutes les TUI. Celle de séquence d'actions, quant à elle est utilisable dans dix-neuf TUI sur vingt à l'exception de l'outil Electronic Blocks [37, 38].

Les autres notions sont moins représentées lors de l'utilisation des TUI. La boucle utilisée dans douze TUI arrive en troisième position. Celle-ci n'est pas automatiquement utilisée chez les jeunes enfants car elle est complexe à comprendre. Par contre, son utilisation est maximisée lorsque le concept semble acquis [35]. Encore, certains enfants essaient d'optimiser leur séquence de programmation en exploitant cette notion. Ensuite, les concepts de « structures conditionnelles » et de « variables » sont représentés à six reprises. Pourtant, « la variable » est une notion clé dans la programmation du langage informatique. Enfin, la notion la moins évoquée est « l'appel de sous-programmes ». Son utilisation n'est possible que dans cinq des vingt outils présentés.

Il est important de relever que l'utilisation d'un concept de programmation ne veut pas nécessairement dire que celui-ci est compris et assimilé. Un concept peut être considéré comme assimilé lorsque l'enfant essaye d'optimiser son « code » en utilisant le concept de boucle, par exemple lors de l'utilisation de T-Maze. Il reste, cependant, complexe de démontrer que les TUI permettent une meilleure perception et assimilation des concepts de programmation.

Toutefois, certains outils mettent en place différentes techniques afin d'apporter, aux utilisateurs, une meilleure compréhension des concepts lors de l'utilisation des TUI. Par exemple, T-Butterfly [21] où la représentation physique de la séquence d'actions via les blocs tangibles est identique au déplacement que l'avatar va effectuer dans le jeu. De même que Tern, qui essaye de mettre en place la représentation physique du concept de la boucle. Aussi, Time de train [5] (TUI non décrite dans les fiches descriptives), met en place un système de correction inédit. L'utilisateur doit reproduire lui-même la séquence de programmation qu'il a créée pour vérifier si sa solution est correcte. Par après, il lui est donné la possibilité de le reproduire sur un ordinateur. Le « Debugging » étape par étape est un autre aspect didactique mis en place par certaines TUI. Via cette technique, l'utilisateur peut découvrir plus facilement où se situe ses erreurs de programmation.

La partie interaction du tableau de critères (Figure 6.3) permet de mettre en évidence, grâce aux critères « tout-en-un » et « composé », que les systèmes sont principalement de type « composé » : dix-huit outils sur vingt, à l'instar de T-maze et TanPro-Kit qui

proposent un plateau de jeu, des blocs de programmation et un résultat affiché sur un écran. Le type « tout-en-un » est présent dans seulement deux des vingt TUI. Celles-ci sont Bee-bot et Electronic bloc, si l'on considère les différents blocs à emboîter comme un seul ensemble et non comme des structures indépendantes. Ces deux TUI sont des TUI proposées à un public plus jeune. Cette théorie rejoindrait celle émise plus tôt montrant qu'afin de faciliter l'utilisation des TUI pour les plus jeunes, les chercheurs utiliseraient des TUI physiques embarquées "tout-en-un".

Une autre constatation est que la majorité des TUI sont de type interaction « passive » (dix-sept TUI sur vingt). L'utilisateur réalise des séquences de blocs de programmation capturées par le biais d'une prise de photo caméra (Turtan, T-Maze) ou par le biais de technologies d'identification embarquée dans les inputs (E-block).

L'interaction « passive via caméra » ressort comme étant la plus utilisée pour ce type d'interaction. Pourtant elle est aussi celle qui pose le plus facilement des problèmes d'utilisations. (problème d'occultation de l'objectif, de placement de la séquence par rapport à la caméra, etc. [12, 13]).

Quelques TUI ajoutent, en plus de celle susmentionnée, d'autres types d'interactions. Par exemple, les TUI T-Maze, E-block, Pléo et Toque qu'outre l'interaction passive utilisent l'interaction discrète par la simple pression d'un bouton. Pour les deux premières citées, cela passe par l'activation de senseurs et pour les deux dernières, cela passe par une interaction via des boutons virtuels tactiles. Aussi, les TUI Starloop et Turtan utilisent en plus de l'interaction passive, l'interaction continue. L'utilisateur doit déposer les blocs tangibles sur une table interactive afin de réaliser leurs séquences de programmation.

Il est important de préciser que toutes les TUI n'ont pas forcément plusieurs types d'interactions. Bee-bot est construit sur base d'une interaction utilisateur par mouvement de type « discrète », la programmation de sa séquence de déplacement se faisant par le biais des boutons placés sur son dos.

Encore, T-ProRob est simplement de type « passive » : pour jouer au jeu, l'utilisateur doit juste réaliser une séquence d'actions et la lire avec une caméra.

Pour terminer cette partie observant les modalités d'interactions entre l'utilisateur et l'input, il existe trois dispositifs proposant une interaction par mouvement de type « embarqué » : Pleo, Electronics blocks et Toque. Ces trois outils sont d'interactions embarquées car chaque input (blocs de programmation) est composé d'un système d'identifiant personnel et de différents capteurs internes sonores, photosensibles etc. Pour les TUI Pléo et Toque, ces capteurs internes permettent, en plus d'avoir des réactions individuelles face à l'environnement, de pouvoir personnaliser certaines ses actions. Ce dernier point permet de mettre en valeur le critère final du tableau. Par ce biais, l'utilisateur a la possibilité de paramétrer les actions et les réactions de certains inputs.

Au niveau des modalités de l'interaction output-utilisateur, il est important de noter que toutes les TUI proposent à l'utilisateur une interaction visuelle. Celles-ci affichent le résultat via le déplacement d'un robot, via l'utilisation de LED ou encore via l'affichage du résultat sur un écran. En revanche, l'utilisation des deux autres modes d'interaction est encore peu utilisée dans le domaine des TUI. Cela peut se remarquer par le fait que les effets sonores ne sont utilisés que par huit TUI sur vingt et que l'interaction

haptique n'est utilisée que par une seule des vingt TUI présentées. De plus, ces deux modes d'interaction ne sont jamais utilisés seuls, ils sont toujours couplés avec un mode d'interaction visuelle.

Chapitre 7

Conclusion

Cette conclusion va être divisée en quatre parties permettant de répondre à la question de recherche émise en début de mémoire : « Les interfaces tangibles utilisateur permettent-elles l'apprentissage des concepts de programmation chez les jeunes ? ». Les trois premiers points vont permettre de répondre à chacune des sous-questions, chaque réponse permettant d'apporter un élément de conclusion à la question principale.

Premièrement, la recherche et la description des différents outils existants dans le domaine de l'interface tangible utilisateur pour l'apprentissage de concepts de programmation, ainsi que la classification originale que j'ai proposée ont mis en avant qu'au niveau didactique ces interfaces permettent d'aborder pas moins de six concepts de programmation : l'algorithme, la séquence, la structure conditionnelle, la boucle, la variable et le sous-programme. La maîtrise de ces six concepts repris dans les cours « d'introduction à la programmation » constitue une base solide pour l'apprentissage de la programmation.

Deuxième, j'ai observé, grâce à la mise en place du tableau de critères originaux, les différentes méthodes d'interaction privilégiées entre l'utilisateur et les interfaces tangibles. Une première méthode d'interaction majoritairement utilisée, entre les inputs et les utilisateurs, est l'interaction dite « passive ». Celle-ci peut, toutefois, être couplée avec d'autres types d'interactions, comme l'interaction par mouvement, mais cela ne représente qu'une minorité des cas.

Pour les interactions entre les outputs et les utilisateurs, la méthode dominante est l'interaction visuelle. Néanmoins, dans une faible proportion, celle-ci peut être couplée d'une interaction sonore ou haptique.

Cette constatation met en avant le peu d'interactions différentes utilisées actuellement dans les interfaces tangibles. Cependant, ce manque n'a pas l'air d'affecter directement la motivation ou l'envie de l'enfant face à l'utilisation d'une interface tangible. Cependant, j'envisagerais des innovations à ce niveau en élargissant les types d'interactions utilisés afin d'augmenter encore plus l'attention, la concentration et l'implication de l'enfant lors de la prise en main des interfaces tangibles.

Troisièmement, mon analyse sur les types de métaphores utilisées pour représenter

les inputs dans les différents dispositifs tangibles selon Fishkin m'a permis de prendre conscience que les interfaces tangibles utilisateur actuelles se contentent d'utiliser une seule des quatre métaphores existantes : la métaphore de Verbe. Seul un dispositif présente une autre métaphore, celle du Verbe & Nom.

Cette observation permet d'ouvrir les portes à la recherche de futurs dispositifs utilisant des métaphores encore peu ou pas exploitées dans le but d'apporter une dimension supplémentaire à l'utilisation des interfaces tangibles.

Pour terminer, les interfaces tangibles utilisateur permettent un accès aisé à l'apprentissage de la programmation. En effet, les enfants ont montré de l'intérêt et de la motivation lors des différentes séances de jeu. Mais le peu d'études réalisées jusqu'à présent ne permettent pas de quantifier les apports de l'utilisation des interfaces tangibles utilisateur sur la maîtrise des différents concepts de programmation.

De plus, au vu du peu de types d'interactions et de métaphores d'inputs utilisés actuellement, en élargissant ceux-ci, on permettrait, certainement, d'apporter un aspect encore plus immersif et attractif pour l'enfant à l'utilisation des interfaces tangibles utilisateur.

Je conclus donc qu'il est impossible dans l'état actuel des recherches de pouvoir affirmer ou infirmer que l'utilisation des interfaces tangibles utilisateur permet l'apprentissage des concepts de programmation chez les jeunes. Malgré cette inconnue, je pense, quand même, que ces outils sont innovateurs dans le domaine en milieu scolaire. Si l'assimilation des concepts de programmation n'a pas encore été mise en lumière, les interfaces tangibles utilisateur apportent déjà énormément en termes de concentrations, motivations et implications de l'enfant qui sont tous trois de bons points de départ à un début d'apprentissage.

Chapitre 8

Travaux futurs

L'apprentissage des concepts de programmation avec les interfaces tangibles utilisateur n'en étant qu'à son commencement. De nombreuses améliorations et modifications peuvent encore être apportées dans ce domaine. En voici quelques uns résultants de ma réflexion :

- Pour l'aspect didactique, un agrandissement du catalogue de concepts de programmation qu'un TUI peut aborder pour essayer de le faire correspondre aux concepts de programmation de base vu dans les cours de hautes écoles ou université.

Et une autre piste serait une mise en avant de la notion de variable car celle-ci est encore trop peu utilisée dans les TUI actuelles.

Il pourrait aussi être intéressant de coupler l'apprentissage de concepts de programmation via les TUI avec l'apprentissage d'autres matières comme les mathématiques ou le français.

Exemple : Mise en place d'une structure conditionnelle sous forme d'un calcul mathématique de type addition , soustraction pour des jeunes enfants de 6 ans. Si le calcul est correct, le robot continue son chemin, sinon il s'arrête et l'enfant doit ré tenter sa chance.

- D'un point de vue plus technique, des améliorations devraient être apportées aux niveaux des interactions de certains outils afin de limiter un maximum les erreurs aux cours des parties de jeux.

Exemple :Revoir la technologie des capteurs externes de type « caméra » permettant la lecture des séquences d'action. Ces technologies génèrent des erreurs et elles ne sont pas facilement manipulables par les enfants. Il faudrait donc travailler avec des technologies embarquées offrant une communication directe avec le système.

L'ajout de la réalité augmentée pourrait être, aussi, une autre amélioration technologique dans les TUI d'apprentissage. Comme le démontre l'exemple de Magicstory Cube, la réalité augmentée apporte une dimension supplémentaire à l'immersion

de l'enfant dans le jeu. Cela lui confère donc une plus grande concentration, une plus grande écoute, une plus grande envie d'apprendre.

De plus, un élargissement des méthodes d'interactions entre l'utilisateur et les inputs et outputs permettraient d'augmenter l'aspect attractif du jeu.

Exemple : Apporter une dimension sonore au niveau des interactions output/utilisateur lors d'erreurs ou alors l'utilisation de boutons physiques lors des interactions input/utilisateur permettant de renforcer l'attention et l'intérêt de l'utilisateur

- Une autre piste dans le domaine des interactions utilisateur et input, pourrait être l'introduction de la gestuelle dans l'utilisation d'une interface tangible, cela permettrait à l'enfant de mettre un mouvement sur un concept utilisé et, par ce fait, de mieux comprendre le concept qu'il utilise [26]. *Exemple : Lors de l'utilisation d'une boucle dans la phase de programmation d'une séquence, l'enfant devrait faire le geste de la boucle avec sa main ou son bras afin de bien comprendre le concept.*
- D'un point de vue étude, il faudrait mettre en place des études permettant d'affirmer ou d'infirmer que l'utilisation des TUI chez les jeunes permet la perception et l'assimilation des concepts de programmation. Cette étude pourrait être réalisée à l'aide d'un prototype dans des classes maternelles ou primaires de la région au cours de plusieurs sessions à répétition. Il est important de dire que des études plus sérieuses sur le sujet commencent à apparaître, des chercheurs en ont d'ailleurs menées une sur base de questionnaires à choix multiples que les enfants devaient compléter. [29]

Bibliographie

- [1] Diana Africano, Sara Berg, Kent Lindbergh, Peter Lundholm, Fredrik Nilbrink, and Anna Persson. Designing tangible interfaces for children's collaboration. In *CHI'04 extended abstracts on Human factors in computing systems*, pages 853–868. ACM, 2004.
- [2] Kunal Chawla, Megan Chiou, Alfredo Sandes, and Paulo Blikstein. Dr. wagon : a'stretchable'toolkit for tangible computer programming. In *Proceedings of the 12th international conference on interaction design and children*, pages 561–564. ACM, 2013.
- [3] Jérôme Delécraz. La théorie de piaget : les stades du développement cognitif de l'enfant, est-ce que votre enfant se développe selon son âge? Url <https://blog.cognifit.com/fr/theorie-de-piaget/>, 04 février 2018.
- [4] Kenneth P Fishkin. A taxonomy for and analysis of tangible interfaces. *Personal and Ubiquitous Computing*, 8(5) :347–358, 2004.
- [5] Gerald Futschek and Julia Moschitz. Learning algorithmic thinking with tangible objects eases transition to computer programming. In *International Conference on Informatics in Schools : Situation, Evolution, and Perspectives*, pages 155–164. Springer, 2011.
- [6] Daniel Gallardo, Carles F Julia, and Sergi Jorda. Turtan : A tangible programming language for creative exploration. In *Horizontal Interactive Human Computer Systems, 2008. TABLETOP 2008. 3rd IEEE International Workshop on*, pages 89–92. IEEE, 2008.
- [7] Anne Ghysselinckx. Psychologie du développement. Haute école Léonard de Vinci, Année 2008-2009.
- [8] Julie Henry. *L'apprentissage de la programmation par des étudiants novices : étude de l'adequation entre concepts, outils d'apprentissage et besoins des utilisateurs*. PhD thesis, Université de Namur, 2017.
- [9] Henry Julie, Bodart Antoine, Dumas Bruno. Programmation tangible pour les enfants : l'étude de l'existant. 2018.
- [10] Michael S Horn and Robert JK Jacob. Tangible programming in the classroom : a practical approach. In *CHI'06 extended abstracts on Human factors in computing systems*, pages 869–874. ACM, 2006.

- [11] Michael S Horn and Robert JK Jacob. Designing tangible programming languages for classroom use. In *Proceedings of the 1st international conference on Tangible and embedded interaction*, pages 159–162. ACM, 2007.
- [12] Michael S Horn and Robert JK Jacob. Tangible programming in the classroom with tern. In *CHI'07 extended abstracts on Human factors in computing systems*, pages 1965–1970. ACM, 2007.
- [13] Michael S Horn, Erin Treacy Solovey, and Robert JK Jacob. Tangible programming and informal science learning : making tuis work for museums. In *Proceedings of the 7th international conference on Interaction design and children*, pages 194–201. ACM, 2008.
- [14] Felix Hu, Ariel Zekelman, Michael Horn, and Frances Judd. Strawbies : explorations in tangible programming. In *Proceedings of the 14th International Conference on Interaction Design and Children*, pages 410–413. ACM, 2015.
- [15] Vassilis Komis and Anastasia Misirli. Robotique pédagogique et concepts préliminaires de la programmation à l'école maternelle : une étude de cas basée sur le jouet programmable bee-bot. In *Sciences et technologies de l'information et de la communication en milieu éducatif : Analyse de pratiques et enjeux didactiques.*, pages 271–281. Athènes : New Technologies Editions, 2011.
- [16] Sophie Lepreux, Julien Castet, Nadine Couture, Emmanuel Dubois, Christophe Kolski, Sébastien Kubicki, Valérie Maquil, and Guillaume Riviere. Interaction tangible sur table, définitions et modèles. *Journal d'Interaction Personne-Système (JIPS)*, 5(1) :1–21, 2016.
- [17] Philipp Locher. Tangible user interfaces : Classification. In *University of Fribourg, Switzerland [Retrieved February 16, 2006, from http://diuf.unifr.ch/people/lalanned/Seminar/Seminar0506/TUI_classification.pdf]*, 2006.
- [18] Javier Marco, Clara Bonillo, and Eva Cerezo. A tangible interactive space odyssey to support children learning of computer programming. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*, pages 300–305. ACM, 2017.
- [19] Anne Roudaut and Eric Lecolinet. Un espace de classification pour l'interaction sur dispositifs mobiles. In *Proceedings of the 19th Conference on l'Interaction Homme-Machine*, pages 99–106. ACM, 2007.
- [20] Kimiko Ryokai, Michael Jongseon Lee, and Jonathan Micah Breitbart. Children's storytelling and programming with robotic characters. In *Proceedings of the seventh ACM conference on Creativity and cognition*, pages 19–28. ACM, 2009.
- [21] Theodosios Sapounidis and Stavros Demetriadis. Touch your program with hands : qualities in tangible programming tools for novice. In *Informatics (PCI), 2011 15th Panhellenic Conference on*, pages 363–367. IEEE, 2011.
- [22] Theodosios Sapounidis, Stavros Demetriadis, and Ioannis Stamelos. Evaluating children performance with graphical and tangible robot programming tools. *Personal and Ubiquitous Computing*, 19(1) :225–237, 2015.

- [23] Florian Scharf, Thomas Winkler, and Michael Herczeg. Tangicons : algorithmic reasoning in a collaborative game for children in kindergarten and first class. In *Proceedings of the 7th international conference on Interaction design and children*, pages 242–249. ACM, 2008.
- [24] Arnan Sipitakiat and Nusarin Nusen. Robo-blocks : designing debugging abilities in a tangible programming system for early primary school children. In *Proceedings of the 11th International Conference on Interaction Design and Children*, pages 98–105. ACM, 2012.
- [25] Andrew C Smith. Dialando : tangible programming for the novice with scratch, processing and arduino. 2010.
- [26] Amber Solomon, Mark Guzdial, Betsy DiSalvo, and Ben Rydal Shapiro. Applying a gesture taxonomy to introductory computing concepts. In *Proceedings of the 2018 ACM Conference on International Computing Education Research*, pages 250–257. ACM, 2018.
- [27] Amanda Sullivan, Mollie Elkin, and Marina Umaschi Bers. Kibo robot demo : engaging young children in programming and engineering. In *Proceedings of the 14th international conference on interaction design and children*, pages 418–421. ACM, 2015.
- [28] Hideyuki Suzuki and Hiroshi Kato. Algoblock : a tangible programming language, a tool for collaborative learning. In *Proceedings of 4th European Logo Conference*, pages 297–303, 1993.
- [29] Alaaeddin Swidan, Felienne Hermans, and Marileen Smit. Programming misconceptions for school students. In *Proceedings of the 2018 ACM Conference on International Computing Education Research*, pages 151–159. ACM, 2018.
- [30] Sureyya Tarkan, Vibha Sazawal, Allison Druin, Evan Golub, Elizabeth M Bonsignore, Greg Walsh, and Zeina Atrash. Toque : designing a cooking-based programming language for and with children. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2417–2426. ACM, 2010.
- [31] Ishii & Ullmer. 1997.
- [32] W. Vanhoof. Introduction à la programmation. Université de Namur, 2014.
- [33] Fédération Wallonie-Bruxelles. <http://www.pactedexcellence.be/>. 2015.
- [34] Danli Wang, Yunfeng Qi, Yang Zhang, and Tingting Wang. Tanpro-kit : a tangible programming tool for children. In *Proceedings of the 12th International Conference on Interaction Design and Children*, pages 344–347. ACM, 2013.
- [35] Danli Wang, Cheng Zhang, and Hongan Wang. T-maze : a tangible programming tool for children. In *Proceedings of the 10th international conference on interaction design and children*, pages 127–135. ACM, 2011.
- [36] Danli Wang, Yang Zhang, and Shengyong Chen. E-block : A tangible programming tool with graphical blocks. *Mathematical Problems in Engineering*, 2013, 2013.

- [37] Peta Wyeth and Helen C Purchase. Tangible programming elements for young children. In *CHI'02 extended abstracts on Human factors in computing systems*, pages 774–775. ACM, 2002.
- [38] Peta Wyeth and Gordon F Wyeth. Electronic blocks : Tangible programming elements for preschoolers. In *IFIP TC. 13 International Conference on Human-Computer Interaction*, volume 1, pages 496–503. IOC Press, 2001.
- [39] Lesley Xie, Alissa N Antle, and Nima Motamedi. Are tangibles more fun ? : comparing children’s enjoyment and engagement using physical, graphical and tangible user interfaces. In *Proceedings of the 2nd international conference on Tangible and embedded interaction*, pages 191–198. ACM, 2008.
- [40] Junnan Yu and Ricarose Roque. A survey of computational kits for young children. In *Proceedings of the 17th ACM Conference on Interaction Design and Children*, pages 289–299. ACM, 2018.
- [41] Zhiying Zhou, Adrian David Cheok, JiunHorng Pan, and Yu Li. Magic story cube : an interactive tangible interface for storytelling. In *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pages 364–365. ACM, 2004.